

# Citra Tekstur Terbaik Untuk *Gaussian Naïve Bayes* Dengan Interpolasi *Nearest Neighbor*

Irwan Budi Santoso<sup>1</sup>, Shoffin Nahwa Utama<sup>1</sup>, Supriyono<sup>1</sup>

<sup>1</sup> Prodi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia

[Diserahkan: 17 Juli 2023, Direvisi: 22 September 2023, Diterima: 13 Desember 2023]

Penulis Korespondensi: Irwan Budi Santoso (email: irwan@ti.uin-malang.ac.id)

**INTISARI** — Salah satu faktor yang berpengaruh terhadap kinerja *Gaussian naïve Bayes classifier* (GNBC) dalam klasifikasi citra tekstur adalah ukuran (dimensi) citra. Ukuran citra adalah salah satu kriteria citra tekstur terbaik di samping nilai pikselnya. Pada penelitian ini, diusulkan metode untuk mendapatkan ukuran citra tekstur terbaik terhadap GNBC dengan optimalisasi interpolasi *nearest neighbor* (NN). Ukuran citra tekstur terbaik dengan nilai piksel hasil interpolasi tersebut membuat GNBC mampu membedakan citra tekstur pada setiap kelasnya dengan kinerja paling tinggi. Langkah pertama metode usulan tersebut adalah menentukan ukuran citra tekstur untuk pelatihan melalui kombinasi ukuran baris dan kolom pada proses optimalisasi. Pengubahan ukuran (*resizing*) semua citra tekstur asal dengan interpolasi NN adalah langkah penting berikutnya untuk mendapatkan citra tekstur baru. Langkah selanjutnya adalah membangun GNBC berdasarkan citra baru hasil interpolasi dan menentukan akurasi klasifikasinya. Langkah terakhir yaitu memilih ukuran citra tekstur terbaik berdasarkan nilai akurasi klasifikasi terbesar sebagai kriteria pertama dan ukuran citra sebagai kriteria kedua. Evaluasi terhadap metode yang diusulkan tersebut dilakukan menggunakan data citra tekstur dari *dataset* publik CVonline dengan melibatkan beberapa skenario uji coba dan metode interpolasi. Hasil uji coba menunjukkan bahwa pada skenario yang melibatkan lima kelas citra tekstur, GNBC dengan interpolasi NN memberikan nilai akurasi klasifikasi terkecil 89% dan terbesar 100% pada ukuran citra terbaik, masing-masing  $14 \times 32$  dan  $47 \times 42$ . Pada skenario yang melibatkan jumlah kelas kecil hingga besar, GNBC dengan interpolasi NN memberikan akurasi klasifikasi 81,6% - 95%. Dari hasil tersebut, GNBC dengan optimalisasi NN memberikan hasil lebih baik daripada metode interpolasi nonadaptif lainnya (*bilinear*, *bicubic*, dan *lanzcos*) dan *principal component analysis* (PCA).

**KATA KUNCI** — Citra, Tekstur, Interpolasi, *Naïve Bayes*, Nonadaptif, Akurasi.

## I. PENDAHULUAN

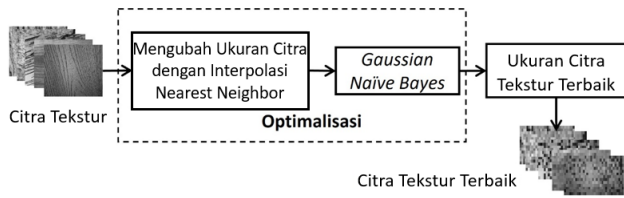
Salah satu kriteria kinerja metode klasifikasi dalam mengklasifikasi objek citra adalah menghasilkan akurasi yang tinggi dalam klasifikasi. Akurasi tinggi yang dihasilkan metode klasifikasi tersebut dipengaruhi oleh beberapa faktor, di antaranya ketepatan dalam mengambil sampel citra objek, praproses citra, ekstraksi fitur citra, dan keandalan metode klasifikasi. Praproses terhadap citra yang biasa dilakukan adalah penskalaan (*scaling*) citra dan memperbaiki kualitas citra dengan filter tertentu. Sementara itu, untuk ekstraksi fitur citra seringkali digunakan transformasi Fourier [1], *wavelet*, dan *principal component analysis* (PCA) [2], [3]. Penskalaan citra biasanya dilakukan dengan pengubahan ukuran (*resizing*) citra asal agar ukuran (dimensi) citra untuk pelatihan dan pengujian menjadi sama. Ukuran citra hasil pengubahan ukuran tersebut selanjutnya menjadi masukan sebuah atau beberapa pengklasifikasi konvensional, seperti *naïve Bayes*. Akan tetapi, untuk ukuran citra yang berbeda, pengklasifikasi dapat memberikan kinerja klasifikasi yang berbeda pula. Kinerja pengklasifikasi tersebut dapat lebih baik atau lebih buruk. Oleh karenanya, mendapatkan ukuran citra yang tepat, sehingga menghasilkan kinerja pengklasifikasi terbaik adalah langkah penting dalam proses klasifikasi.

Tidak banyak penelitian sebelumnya yang fokus pada pengubahan ukuran citra untuk meningkatkan kinerja dalam klasifikasi. Kebanyakan peneliti hanya fokus pada perbaikan citra dengan menggunakan satu atau beberapa metode interpolasi citra [4]–[8]. Dalam klasifikasi citra, pengubahan ukuran citra (misalnya citra tekstur) menjadi persoalan

tersendiri terhadap keberhasilan metode klasifikasi. Fakta penelitian tersebut menunjukkan bahwa jarang sekali para peneliti menggunakan metode interpolasi untuk tujuan mendapatkan ukuran citra terbaik bagi metode klasifikasi serta tidak banyak penelitian menggunakan hasil interpolasi sebagai pendekatan terhadap nilai fitur dari objek citra.

Beberapa penelitian sebelumnya berfokus pada perbaikan metode klasifikasi, di antaranya dengan memodifikasi struktur model atau dengan memperbaiki estimasi parameter (bobot) model. Salah satu metode klasifikasi berbasis probabilitas yang relatif sederhana dan sering digunakan adalah *Gaussian naïve Bayes classifier* (GNBC). Beberapa penelitian terkait GNBC yang fokus pada perbaikan metode telah memodifikasi struktur model [9] dan memperbaiki estimasi parameter (bobot) model [10]–[14]. Dari semua penelitian tersebut, diketahui bahwa secara prinsip GNBC merupakan metode klasifikasi yang tetap mengasumsikan bahwa bahwa atribut (fitur) yang satu bersifat independen terhadap atribut (fitur) yang lain [15]–[17]. Adanya asumsi tersebut membuat metode GNBC mudah untuk digunakan atau diimplementasikan, selalu menjadi topik hangat, dan terus diteliti, terutama terkait cara peningkatan kinerja klasifikasi.

Berpijak dari kelemahan dan kelebihan metode GNBC serta potensi penggunaan metode interpolasi, pada penelitian ini diusulkan metode untuk mendapatkan ukuran citra tekstur terbaik menggunakan interpolasi serta untuk meningkatkan kinerja GNBC dalam klasifikasi citra tekstur. Kontribusi penelitian ini adalah penggunaan metode interpolasi untuk mendapatkan ukuran citra tekstur terbaik pada GNBC,



**Gambar 1.** Proses utama optimalisasi interpolasi *nearest neighbor* untuk mendapatkan citra tekstur terbaik.

peningkatan kinerja GNBC dengan ukuran citra tekstur terbaik menggunakan interpolasi *nearest neighbor* (NN), dan mereduksi ukuran citra tekstur untuk masukan pada GNBC.

## II. METODE YANG DIUSULKAN

Secara garis besar, tahapan dalam penelitian ini meliputi akuisisi citra tekstur untuk pelatihan, optimalisasi ukuran citra dengan metode interpolasi NN terhadap kinerja GNBC, dan mendapatkan ukuran citra tekstur terbaik (optimal) untuk ekstraksi fitur, seperti diperlihatkan pada Gambar 1. Akuisisi citra dijalankan untuk mengambil data citra tekstur per kelas, mendapatkan nilai piksel dari setiap citra, serta mengumpulkan nilai tersebut dalam bentuk matriks [18]. Pada penelitian ini, metode interpolasi digunakan untuk menentukan nilai piksel citra dengan ukuran baru terhadap citra asal. Ukuran citra baru tersebut dibentuk mulai dari ukuran kecil dengan kombinasi baris dan kolom sampai ukuran paling besar, bukan berdasarkan rasio terhadap citra asal. Sementara itu, optimalisasi ukuran citra dengan interpolasi NN pada prinsipnya adalah menjadikan setiap citra tekstur baru hasil perubahan ukuran terhadap citra tekstur asal sebagai masukan pelatihan pada GNBC. Selanjutnya, dilakukan proses untuk mendapatkan ukuran citra tekstur terbaik di antara citra baru yang telah dibentuk. Dari ukuran citra tekstur terbaik tersebut, didapatkan citra tekstur terbaik yang tepat bagi GNBC sedemikian hingga GNBC dapat membedakan citra tekstur setiap kelasnya dengan kinerja terbaik.

### A. NEAREST NEIGHBOR-GAUSSIAN NAÏVE BAYES CLASSIFIER (NN-GNBC)

Interpolasi dengan NN adalah interpolasi paling sederhana, sehingga waktu komputasi metode tersebut relatif lebih cepat daripada metode interpolasi yang lain. Kondisi ini terjadi karena prinsip metode interpolasi tersebut adalah memilih nilai piksel terdekat yang ada di sekelilingnya. Nilai piksel yang baru didapatkan dari nilai piksel terdekat untuk memperkecil ataupun memperbesar citra asal [4]. *Kernel* dari interpolasi NN secara matematis ditulis sebagai berikut.

$$l(x) = \begin{cases} 0 & |x| \geq 0,5 \\ 1 & |x| < 0,5 \end{cases} \quad (1)$$

dengan  $|x|$  adalah jarak antara titik dengan *grid* titik yang akan diinterpolasi pada citra tekstur. Fungsi *kernel* tersebut digunakan untuk mendapatkan nilai piksel hasil interpolasi.

Jika diketahui  $T = \{z_1, \dots, z_d\}$  adalah citra hasil dari interpolasi dengan NN dan dianggap sebagai nilai atribut (fitur) yang meliputi  $z_1, \dots, z_d$ , dengan  $d$  adalah ukuran fitur (baris  $\times$  kolom) citra, maka fitur citra tersebut selanjutnya digunakan untuk membentuk atau membangun GNBC [19]. Dalam pembentukan GNBC tersebut, diasumsikan bahwa di antara atribut (fitur) citra bersifat independen serta diasumsikan juga bahwa setiap fitur atau atributnya memiliki distribusi normal (Gaussian). Selanjutnya, dengan memperhatikan asumsi-asumsi tersebut, nilai probabilitas atribut (fitur), jika kelas ke- $j$  ( $C_j$ ) diketahui, adalah sebagai berikut [17].

$$P(T|C_j) = \prod_{k=1}^d P(T_k|C_j) = \prod_{k=1}^d N(z_k; \hat{\mu}_{jk}, \hat{\sigma}_{jk}) \quad (2)$$

dengan  $\hat{\mu}_{jk}$  dan  $\hat{\sigma}_{jk}$  adalah nilai estimasi parameter rerata dan simpangan baku untuk atribut ke- $k$  dan kelas ke- $j$ . Dengan menggunakan kaidah probabilitas bersyarat, selanjutnya didapatkan persamaan sebagai berikut.

$$P(C_j, T) = P(C_j)P(T|C_j) = P(C_j) \prod_{k=1}^d N(z_k; \hat{\mu}_{jk}, \hat{\sigma}_{jk}). \quad (3)$$

Dari (3), selanjutnya dapat ditentukan hasil klasifikasi berdasarkan atribut (fitur) citra yang dimasukkan dan nilai probabilitas terbesar dari  $P(C_j, T)$ , dalam persamaan berikut.

$$\hat{c} = \underset{c_j}{\operatorname{argmax}} \{P(C_j) \prod_{k=1}^d N(z_k; \hat{\mu}_{jk}, \hat{\sigma}_{jk})\} \quad (4)$$

dengan  $\hat{c}$  adalah hasil klasifikasi.

Untuk mengatasi masalah komputasi, selanjutnya dapat dilakukan operasi logaritma terhadap (3) yang menghasilkan fungsi diskriminan untuk setiap kelasnya, seperti berikut.

$$g_j(T) = \log(P(C_j)) - \sum_{k=1}^d \log(\hat{\sigma}_{jk}) - \frac{1}{2} \sum_{k=1}^d \frac{(z_k - \hat{\mu}_{jk})^2}{\hat{\sigma}_{jk}^2} \quad (5)$$

dengan  $g_j$  adalah fungsi diskriminan kelas ke- $j$ .

Persamaan (1) hingga (5) memperlihatkan keterkaitan antara metode interpolasi NN dengan GNBC, yang selanjutnya pada penelitian ini dinamakan NN-GNBC. Tingkat kinerja GNBC pada kasus ini sangat dipengaruhi oleh fitur hasil interpolasi dan tidak semua hasil interpolasi memberikan tingkat kinerja yang baik. Untuk mendapatkan hasil interpolasi yang memberikan tingkat kinerja yang baik pada GNBC, diperlukan optimalisasi metode interpolasi. Kendati citra yang diperoleh dengan interpolasi NN relatif kurang baik dibandingkan metode interpolasi lain, belum tentu interpolasi tersebut memberikan pengaruh kurang baik terhadap kemampuan GNBC dalam mengklasifikasikan objek citra.

### B. ALGORITMA OPTIMALISASI INTERPOLASI NEAREST NEIGHBOR

Optimalisasi interpolasi NN pada penelitian ini adalah perubahan ukuran terhadap semua citra dalam pelatihan untuk mendapatkan ukuran citra yang optimal. Optimalisasi tersebut dilakukan untuk menemukan ukuran citra tekstur dan nilai interpolasi yang tepat bagi GNBC, sehingga mampu membedakan citra pada setiap kelasnya dengan akurasi paling tinggi. Langkah pertama untuk mendapatkan ukuran citra yang terbaik adalah mengubah ukuran semua citra tekstur untuk pelatihan menggunakan metode interpolasi NN menjadi citra tekstur baru berdasarkan ukuran citra yang telah ditentukan. Kedua, GNBC dibangun berdasarkan masukan citra tekstur baru hasil interpolasi. Ketiga, semua citra tekstur baru hasil interpolasi yang digunakan untuk pelatihan diklasifikasikan. Lalu, langkah pertama sampai ketiga diulangi dengan berbagai kombinasi ukuran baris dan kolom citra tekstur. Keempat, dilakukan pemilihan GNBC yang menghasilkan akurasi tertinggi dengan ukuran citra tekstur paling kecil. Selanjutnya, langkah kelima adalah mendapatkan ukuran citra tekstur dengan akurasi GNBC tertinggi sebagai citra tekstur terbaik, seperti yang ditunjukkan pada Algoritma Optimalisasi-NN-GNBC.

Masukan pada Algoritma Optimalisasi-NN-GNBC adalah *rowIm*, *collm*, *nSample*, *Im*, dan *C*. Masukan *rowIm* merupakan ukuran baris citra pelatihan, sedangkan *collm* merupakan ukuran kolom citra pelatihan, dengan asumsi setiap ukuran citra pelatihan adalah sama. Masukan *nSample* adalah

banyaknya citra yang digunakan dalam pelatihan,  $Im$  adalah citra asal, dan  $C$  adalah kelas atau label dalam setiap citra pelatihan. Sementara itu, keluaran algoritma tersebut adalah  $bestSizeIm$ , yang menunjukkan ukuran citra pelatihan terbaik dan memberikan akurasi klasifikasi GNBC paling tinggi. Seperti pada penjelasan sebelumnya, pada penelitian ini dilakukan perubahan ukuran citra pelatihan dengan berbagai ukuran yang mungkin. Pada algoritma tersebut, perubahan ukuran citra dilakukan mulai dari ukuran baris dan kolom citra pelatihan terkecil ( $1 \times 1$ ) sampai ukuran paling besar ( $rowIm \times colIm$ ). Hasil perubahan ukuran citra untuk setiap ukuran adalah  $resizeIm$ , yang diperoleh dengan fungsi interpolasi NN dengan menggunakan *kernel* interpolasi NN seperti pada (1). Selanjutnya, hasil perubahan setiap ukuran dan yang diberlakukan pada setiap citra pelatihan diubah ke dalam bentuk vektor ( $X$ ) dengan fungsi *reshape*. Dengan menggunakan fungsi *TrainingGNBC*, diperoleh nilai akurasi dalam klasifikasi untuk setiap ukuran citra. Kemudian, setiap hasil dari perubahan ukuran citra dan akurasinya ditampung pada  $rowdim$ ,  $coldim$ , dan  $accuracyGNBC$ . Dari hasil optimalisasi tersebut, selanjutnya dengan menggunakan fungsi *OptimalImageDim*, didapatkan ukuran baris dan kolom citra yang terbaik dengan akurasi GNBC tertinggi dan ditampung pada  $bestSizeIm$ . Ukuran baris dan kolom citra terbaik adalah ukuran dengan perkalian baris dan kolom yang paling kecil dengan akurasi GNBC paling tinggi.

#### Algoritma Optimalisasi-NN-GNBC

//Optimalisasi interpolasi NN untuk mendapat ukuran citra terbaik bagi GNBC

Input: rowIm, colIm, nSample, Im, C

//rowIm: ukuran baris citra, colIm: ukuran kolom citra, nSample: banyaknya //citra dalam training, Im: citra asal dan C: kelas atau label

Output: bestSizeIm

//ukuran baris dan kolom citra yang menghasilkan akurasi terbaik untuk GNBC

iter  $\leftarrow 0$  //inisialisasi

for  $i = 1$  to rowIm do

for  $j = 1$  to colIm do

for  $k = 1$  to nSample do

$resizeIm \leftarrow$  interpolasiNN(Im(k),i,j)

//resize citra dengan kernel NN menggunakan (1)

$X(k,1:i*j) \leftarrow$  reshape (resizeIm,1,i\*j)

end

accuracy  $\leftarrow$  TrainingGNBC(X,C)

//menggunakan (4) dan (5)

iter  $\leftarrow$  iter+1

rowdim(iter)  $\leftarrow i$

coldim(iter)  $\leftarrow j$

accuracyGNBC(iter)  $\leftarrow$  accuracy

end

end

bestSizeIm  $\leftarrow$  OptimalImageDim(rowdim, coldim, accuracyGNBC)

Variabel  $bestSizeIm$  diperoleh dengan menggunakan fungsi *OptimalImageDim* yang prosesnya dilakukan menggunakan algoritma Optimal-Image-Dim. Secara prinsip, algoritma tersebut berjalan dengan menemukan nilai akurasi GNBC paling tinggi ( $bestaccuracy$ ) dari berbagai ukuran citra pelatihan hasil perubahan ukuran dengan interpolasi. Akan tetapi, nilai akurasi yang paling tinggi dapat diperoleh pada beberapa ukuran citra, sehingga jika ada beberapa nilai akurasi tertinggi yang sama, kriteria yang digunakan adalah ukuran citra hasil perubahan ukuran paling kecil. Untuk memeriksa adanya nilai akurasi tinggi yang sama, dilakukan operasi

$accuracyGNBC(i) = bestaccuracy$ , sedangkan untuk memperoleh ukuran citra paling kecil, digunakan fungsi *bindex*, yaitu indeks perkalian ukuran baris dengan ukuran kolom citra ( $bsize$ ) yang paling kecil nilainya. Selanjutnya, dari indeks tersebut didapatkan ukuran citra yang paling optimal ( $bestrowsize$ ,  $bestcolsize$ , dan  $bestaccuracy$ ).

#### Algoritma Optimal-Image-Dim

//Fungsi mendapatkan ukuran citra terbaik

Input: rowdim, coldim, accuracyGNBC

Output: bestrowsize, bestcolsize, bestaccuracy

//  $bestrowsize$ : ukuran baris terbaik,  $bestcolsize$ : ukuran kolom citra terbaik

//  $bestaccuracy$ : akurasi terbaik untuk GNBC

$j \leftarrow 0$  //inisialisasi

bestaccuracy  $\leftarrow$  max(accuracyGNBC)

for  $i = 1$  to rowdim\*coldim do

if (accuracyGNBC(i) = bestaccuracy)

$j \leftarrow j + 1$

$browsize(j) \leftarrow$  rowdim(i)

$bcolsize(j) \leftarrow$  coldim(i)

$bsize(j) \leftarrow$  browsize(j)\*bcolsize(j)

end

end

bindex  $\leftarrow$  min(bsize)

bestrowsize  $\leftarrow$  browsize(bindex)

bestcolsize  $\leftarrow$  bcolsize(bindex)

### C. ALGORITMA PENCARIAN FITUR TERBAIK DARI HASIL OPTIMALISASI INTERPOLASI NN

Apabila ukuran citra terbaik beserta nilai pikselnya yang diperoleh dari optimalisasi interpolasi NN dianggap mewakili fitur utama citra, maka nilai hasil ekstraksi fitur citra sama dengan nilai hasil interpolasi citra pada ukuran citra terbaik tersebut. Nilai ekstraksi fitur tersebut selanjutnya ditampung dalam vektor, seperti yang ditunjukkan pada Algoritma fitur-NN. Nilai ekstraksi fitur tersebut akan berpengaruh terhadap struktur model GNBC yang dibuat, hasil estimasi parameter model, dan klasifikasi. Pada Algoritma fitur-NN, nilai ekstraksi fitur adalah  $bestresize$ , yang diperoleh berdasarkan ukuran citra terbaik ( $bestrowsize$  dan  $bestcolsize$ ) dan fungsi *interpolasiNN*. Hasil interpolasi citra tersebut selanjutnya diubah ke dalam bentuk vektor  $y$ .

#### Algoritma fitur-NN

//Ekstraksi fitur untuk GNBC berdasarkan hasil optimalisasi interpolasi NN

Input: bestrowsize, bestcolsize, Im

Output:  $y$

//fitur citra terbaik untuk GNBC hasil optimalisasi NN

bestresize  $\leftarrow$  interpolasiNN (Im, bestrowsize, bestcolsize)

$y$  (1: bestrowsize\*bestcolsize)  $\leftarrow$  reshape (bestresize, 1, bestrowsize\* bestcolsize)

### III. SKENARIO UJI COBA

Untuk mengetahui kinerja metode yang diusulkan, pada penelitian ini dilaksanakan uji coba dengan beberapa skenario. Uji coba tersebut dilakukan dengan menggunakan data publik yang telah disediakan oleh Lazebnik, Schmid, dan Ponce, yang dapat diakses di CVonline: Image Databases, yaitu berupa *dataset* citra tekstur [20]. *Dataset* tersebut terdiri atas 25 kelas/label, yang pada setiap kelasnya terdapat 40 sampel citra tekstur, sehingga jumlah sampel keseluruhan adalah seribu citra tekstur (*grayscale*). Setiap citra pada *dataset* tersebut memiliki resolusi yang sama, yaitu  $480 \times 640$  piksel. Dari *dataset* citra tersebut, disusun beberapa kelompok *dataset* dan beberapa skenario uji coba seperti disajikan pada Tabel I dan Tabel II.

TABEL I  
DESKRIPSI DATASET SKENARIO 1: CITRA TEKSTUR (LIMA KELAS)

Citra	Kelas	Contoh
T01-T05	bark1-bark2- bark3-wood1- wood2	
T06-T10	wood3- water-granite- marble-floor1	
T11-T15	floor2-pebbles-wall-brick1-brick2	
T16-T20	glass1-glass2-carpet1-carpet2-uphostery	
T21-T25	wallpaper-fur-knit-corduroy-plaid	

TABEL II  
DESKRIPSI DATASET SKENARIO 2: CITRA TEKSTUR (2-25 KELAS)

Citra Tekstur	Banyak Kelas	Jumlah Sampel
T01-T02	2	11
T01-T03	3	13
T01-T05	5	17
T01-T07	7	21
T01-T09	9	25

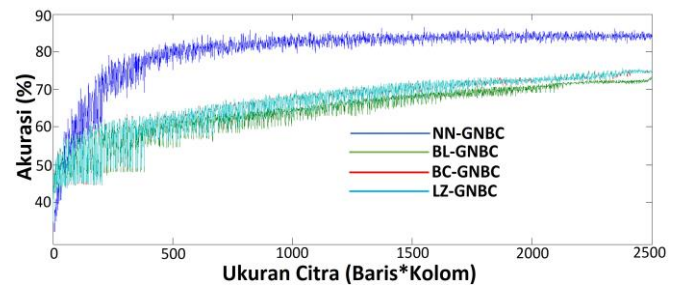
Skenario 1 menggunakan dataset pada Tabel I yang melibatkan lima kelas/label, meliputi citra tekstur T01-T05, citra tekstur T06-T10, citra tekstur T11-T15, citra tekstur T16-T20, dan citra tekstur T21-T25. Pada setiap dataset skenario tersebut, dilakukan optimalisasi interpolasi NN, sehingga diperoleh ukuran citra pelatihan yang optimal terhadap GNBC. Uji coba pada skenario 1 bertujuan untuk mengetahui secara lebih mendalam kinerja metode yang diusulkan dengan melihat nilai akurasi (*accuracy*), presisi (*precision*), sensitivitas (*recall*) dan *F-score* [21], [22].

Skenario berikutnya, yaitu Skenario 2, melibatkan dataset pada Tabel II. Proses uji coba hampir sama dengan uji coba pada Skenario 1, hanya saja data pada Skenario 2 digunakan untuk mengetahui konsistensi kinerja metode yang diusulkan. Oleh karena itu, pada skenario tersebut diujicobakan dataset citra pelatihan dengan jumlah kelas kecil (dua kelas/label), yaitu (T01-T02), sampai dengan citra pelatihan dengan kelas besar (25 kelas/label), yaitu (T01-T25).

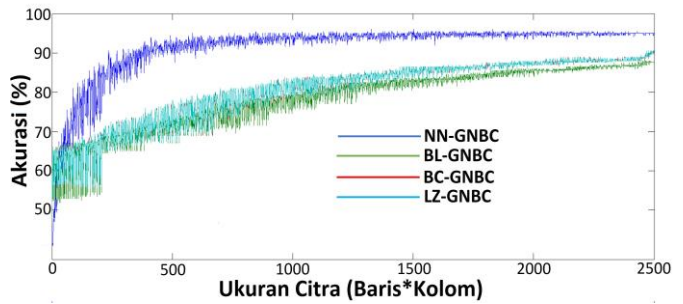
Untuk mengetahui tingkat keberhasilan optimalisasi interpolasi NN terhadap kinerja GNBC, hasil optimalisasi yang dibuat ini dibandingkan juga dengan hasil optimalisasi yang menggunakan metode interpolasi nonadaptif lain, yaitu *bilinear*, *bicubic*, dan Lanczos, serta optimalisasi dengan metode PCA [3], [23]. Metode *bilinear* dan *bicubic* pada penelitian sebelumnya menghasilkan citra yang lebih tajam daripada NN [24]. Pada metode *bicubic*, secara umum, biasanya nilai *a* ditentukan dari nilai -0,5 sampai -0,75 [25]–[27]. Dalam penelitian ini, digunakan nilai *a* = -0,5. Untuk metode interpolasi Lanczos [6], [8], [28], [29], dalam penelitian ini ditentukan nilai *a* berupa integer positif 2 (*a* = 2) atau disebut juga interpolasi Lanczos orde 2.

Untuk mempermudah dalam membandingkan metode-metode yang diujicobakan dalam setiap skenario, selanjutnya dilakukan penamaan terhadap beberapa perlakuan metode yang diujicobakan sebagai berikut.

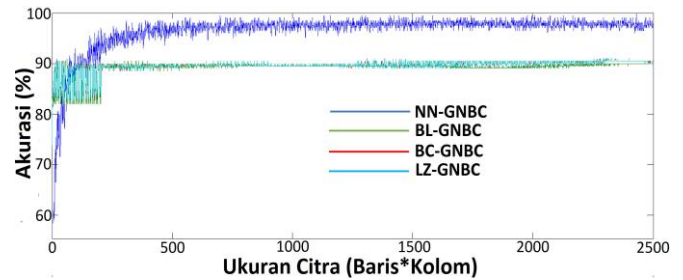
- Pengubahan ukuran citra tekstur dengan interpolasi NN untuk mendapatkan ukuran citra pelatihan terbaik terhadap GNBC dinamakan NN-GNBC.



(a)



(b)



(c)

Gambar 2. Contoh proses optimalisasi untuk mendapatkan ukuran citra terbaik terhadap GNBC pada skenario 1. (a) T01-T05, (b) T06-T10, (c) T16-T20.

- Pengubahan ukuran citra tekstur dengan interpolasi *bilinear* untuk mendapatkan ukuran citra pelatihan terbaik terhadap GNBC dinamakan BL-GNBC.
- Pengubahan ukuran citra tekstur dengan interpolasi *bicubic* untuk mendapatkan ukuran citra pelatihan terbaik terhadap GNBC dinamakan BC-GNBC.
- Pengubahan ukuran citra tekstur dengan interpolasi Lanczos orde 2 (*a* = 2) untuk mendapatkan ukuran citra pelatihan terbaik terhadap GNBC dinamakan LZ-GNBC.
- Pengubahan ukuran citra tekstur dengan interpolasi NN dilanjutkan dengan ekstraksi fitur menggunakan PCA berdasarkan *eigenvalue* ( $\lambda$ ) yang terpilih terhadap GNBC dinamakan NN-PCA-GNBC.

#### IV. HASIL UJI COBA DAN PEMBAHASAN

##### A. SKENARIO 1

Uji coba pada Skenario 1 memberikan hasil sebagai berikut. Pertama, hasil pada proses optimalisasi pengubahan ukuran citra tekstur dengan metode interpolasi berdasarkan nilai akurasi klasifikasi GNBC. Lalu, hasil kedua adalah ukuran citra tekstur terbaik yang didapatkan dari proses optimalisasi. Hasil uji coba yang ketiga adalah hasil pengukuran kinerja GNBC berdasarkan masukan citra tekstur terbaik dengan melihat nilai akurasi, presisi, sensitivitas, dan *F-score*. Hasil uji coba dalam setiap skenario dibatasi ukuran maksimum baris dan kolom citra dalam iterasi, yaitu 50 dan 50 (50 × 50). Oleh karenanya,

TABEL III

UKURAN CITRA TERBAIK HASIL OPTIMALISASI PADA SKENARIO 1 DAN 2

Dataset Citra Tekstur	$\lambda$ PCA*	Ukuran Citra Tekstur Terbaik				
		NN-GNBC	BL-GNBC	BC-GNBC	LZ-GNBC	
Skenario 1	T01-T05	$\lambda_{1, \dots, 36}$	14 × 32	47 × 49	41 × 50	41 × 50
	T06-T10	$\lambda_{1, \dots, 38}$	20 × 27	40 × 50	46 × 48	45 × 48
	T11-T15	$\lambda_{1, \dots, 39}$	47 × 8	46 × 50	40 × 48	40 × 48
	T16-T20	$\lambda_{1, \dots, 37}$	47 × 42	4 × 5	3 × 3	22 × 25
	T21-T25	$\lambda_{1, \dots, 36}$	4 × 35	27 × 36	24 × 19	23 × 20
Skenario 2	T01-T02	$\lambda_{1, \dots, 36}$	9 × 17	2 × 2	17 × 4	44 × 6
	T01-T03	$\lambda_{1, \dots, 36}$	8 × 21	37 × 9	14 × 11	47 × 6
	T01-T05	$\lambda_{1, \dots, 36}$	14 × 32	47 × 49	41 × 50	41 × 50
	T01-T07	$\lambda_{1, \dots, 38}$	30 × 19	47 × 49	41 × 50	41 × 50
	T01-T09	$\lambda_{1, \dots, 38}$	41 × 17	49 × 50	41 × 50	45 × 40
	T01-T11	$\lambda_{1, \dots, 38}$	30 × 19	43 × 48	41 × 50	41 × 50
	T01-T13	$\lambda_{1, \dots, 39}$	26 × 27	49 × 50	50 × 50	50 × 50
	T01-T17	$\lambda_{1, \dots, 37}$	26 × 35	50 × 50	45 × 50	45 × 50
	T01-T21	$\lambda_{1, \dots, 39}$	26 × 46	50 × 50	50 × 50	45 × 50
T01-T25	$\lambda_{1, \dots, 37}$	34 × 36	49 × 50	46 × 50	49 × 50	

\*) citra yang digunakan untuk PCA adalah 10% dari ukuran asal

TABEL IV

AKURASI GNBC HASIL OPTIMALISASI METODE INTERPOLASI PADA SKENARIO 1 DAN 2

Dataset Citra Tekstur	Akurasi (%)					
	NN-PCA-GNBC	NN-GNBC	BL-GNBC	BC-GNBC	LZ-GNBC	
Skenario 1	T01-T05	63,00	89,00	78,50	79,00	79,00
	T06-T10	76,50	96,00	88,00	90,50	90,50
	T11-T15	49,00	93,00	84,50	86,00	86,00
	T16-T20	69,50	100	91,00	91,00	91,00
	T21-T25	62,50	94,00	89,00	89,00	89,50
Skenario 2	T01-T02	81,25	95,00	92,50	92,50	93,75
	T01-T03	58,33	93,33	89,17	88,33	89,17
	T01-T05	63,00	89,00	78,50	79,00	79,00
	T01-T07	58,93	88,57	72,86	73,93	73,93
	T01-T09	57,50	86,67	73,33	75,56	75,56
	T01-T11	51,14	85,00	71,14	73,64	73,64
	T01-T13	46,92	86,15	72,50	75,00	75,00
	T01-T17	41,32	82,65	67,21	69,71	69,71
	T01-T21	36,31	82,02	69,88	71,79	71,79
T01-T25	28,60	81,60	67,20	68,80	68,90	

\*) citra yang digunakan untuk PCA adalah 10% dari ukuran asal

banyaknya kombinasi ukuran citra tekstur dalam proses optimalisasi adalah 2.500. Proses iterasi tersebut dilakukan untuk memperoleh citra tekstur terbaik. Kriteria terbaik tersebut didasarkan pada nilai akurasi klasifikasi terbaik sebagai prioritas pertama dan ukuran citra terkecil sebagai prioritas berikutnya.

Untuk skenario 1 dengan dataset citra tekstur T01-T05, proses iterasi dalam optimalisasi untuk memperoleh ukuran citra terbaik ditunjukkan pada Gambar 2(a). Dari gambar tersebut, terlihat bahwa ukuran (baris × kolom) < 15 NN-GNBC secara umum menghasilkan nilai akurasi lebih kecil dibandingkan metode yang lain. Akan tetapi, citra dengan ukuran (baris × kolom) ≥ 15 NN-GNBC memberikan nilai akurasi lebih baik daripada BL-GNBC, BC-GNBC, dan LZ-GNBC. Metode BC-GNBC dan LZ-GNBC memberikan hasil relatif sama dan masih lebih baik dibandingkan dengan metode BL-GNBC. Kemudian, untuk dataset T06-T10, Gambar 2(b) menunjukkan bahwa NN-GNBC secara umum juga

TABEL V

CONTOH HASIL INTERPOLASI NN PADA CITRA T01-T05

Citra Tekstur	Citra Asal (480 × 640)	Citra Hasil Interpolasi NN (14 × 32)
T01		
T02		
T03		
T04		
T05		

menghasilkan akurasi lebih kecil dibandingkan metode yang lain pada ukuran (baris × kolom) < 21 dan lebih baik daripada metode yang lain pada ukuran (baris × kolom) ≥ 21. Demikian juga untuk dataset T16-T20 pada Gambar 2(c), NN-GNBC dengan ukuran (baris × kolom) < 41 secara umum memperoleh akurasi lebih rendah daripada yang lain, tetapi ukuran (baris × kolom) ≥ 41 lebih baik daripada yang lain. Sementara itu, untuk dataset T11-T15 dan T21-T25, proses optimalisasi memperlihatkan hasil yang hampir sama seperti pada dataset sebelumnya, yaitu NN-GNBC memperoleh akurasi lebih baik daripada BL-GNBC, BC-GNBC, dan LZ-GNBC pada ukuran citra (baris × kolom) ≥ 49 dan (baris × kolom) ≥ 57.

Berdasarkan proses optimalisasi ukuran citra pelatihan pada data Skenario 1, dengan dataset citra T1-T5, T06-T10, T11-T15, T16-T20, dan T21-T25, diperoleh nilai akurasi klasifikasi tertinggi beserta ukuran citra pelatihan seperti pada Tabel III dan Tabel IV. Hasil uji coba tersebut menunjukkan bahwa NN-GNBC secara keseluruhan memiliki akurasi lebih baik dalam klasifikasi citra tekstur dibandingkan dengan BL-GNBC, BC-GNBC, dan LZ-GNBC, dengan ukuran citra terbaik (optimal) berbeda-beda pada setiap dataset. Pada uji coba dengan dataset T01-T05, hasil optimalisasi menunjukkan bahwa NN-GNBC terbaik dihasilkan pada ukuran citra 14 × 32 dengan akurasi 89%, sedangkan BL-GNBC terbaik pada ukuran citra 47 × 49 dengan akurasi 78,5%, BC-GNBC terbaik pada ukuran citra 41 × 50 dengan akurasi 79%, serta LZ-GNBC terbaik pada ukuran citra 41 × 50 dengan akurasi 79%. Hasil tersebut memperlihatkan bahwa NN-GNBC memberikan hasil paling baik dari sisi akurasi maupun ukuran citra (terkecil). Untuk percobaan T06-T10, T11-T15, dan T21-T25, secara umum diperoleh hasil yang sama, yaitu NN-GNBC terbaik dari sisi akurasi ataupun ukuran citra (terkecil), sedangkan untuk percobaan dengan dataset T16-T20, kendati ukuran citra terbaik yang diperoleh NN-GNBC masih lebih besar daripada metode lainnya, nilai akurasinya masih yang terbaik.

Untuk pembandingan metode ekstraksi fitur lain, yaitu metode PCA, uji coba diawali dengan mereduksi citra asal dengan ukuran 480 × 640 piksel menjadi 48 × 64 menggunakan interpolasi NN. Uji coba hanya menggunakan NN karena NN memberikan kinerja terbaik terhadap GNBC di antara metode interpolasi yang lain. Ekstraksi fitur citra dengan PCA dilakukan dengan beberapa eigenvalue. Eigenvalue yang digunakan beserta akurasi terbaiknya ditunjukkan pada Tabel III dan Tabel IV. Hasil uji coba tersebut menunjukkan bahwa ekstraksi fitur menggunakan PCA dengan GNBC (NN-PCA-GNBC) mendapatkan hasil lebih buruk daripada metode lainnya.

Secara umum, hasil uji coba pada Skenario 1 menunjukkan bahwa NN-GNBC menghasilkan akurasi terbaik dibandingkan metode yang lain. Hasil tersebut tidak dapat dilepaskan dari keberhasilan interpolasi NN dalam melakukan perubahan

TABEL VI  
PRESISI, SENSITIVITAS, DAN *F-SCORE* HASIL UJI COBA PADA SKENARIO 1

Citra	Presisi (%)				Sensitivitas (%)				<i>F-score</i> (%)			
	NN-GNBC	BL-GNBC	BC-GNBC	LZ-GNBC	NN-GNBC	BL-GNBC	BC-GNBC	LZ-GNBC	NN-GNBC	BL-GNBC	BC-GNBC	LZ-GNBC
T01	97,06	85,71	85,71	85,71	82,50	75,00	75,00	75,00	89,19	80,00	80,00	80,00
T02	80,00	73,08	73,58	73,58	100	95,00	97,50	97,50	88,89	82,61	83,87	83,87
T03	100	100	100	100	87,50	85,00	85,00	85,00	93,33	91,89	91,89	91,89
T04	82,61	68,29	68,29	68,29	95,00	70,00	70,00	70,00	88,37	69,14	69,14	69,14
T05	91,43	71,05	72,97	72,97	80,00	67,50	67,50	67,50	85,33	69,23	70,13	70,13
T06	100	97,22	100	100	97,50	87,50	90,00	90,00	98,73	92,11	94,74	94,74
T07	100	95,00	95,12	95,12	100	95,00	97,50	97,50	100	95,00	96,30	96,30
T08	100	77,55	82,22	82,22	100	95,00	92,50	92,50	100	85,39	87,06	87,06
T09	100	92,11	97,22	97,22	82,50	87,50	87,50	87,50	90,41	89,74	92,11	92,11
T10	83,33	81,08	80,95	80,95	100	75,00	85,00	85,00	90,91	77,92	82,93	82,93
T11	100	93,94	97,06	97,06	87,50	77,50	82,50	82,50	93,33	84,93	89,19	89,19
T12	85,11	86,36	88,37	88,37	100	95,00	95,00	95,00	91,95	90,48	91,57	91,57
T13	100	97,44	100	100	100	95,00	95,00	95,00	100	96,20	97,44	97,44
T14	90,91	64,91	65,52	65,52	100	92,50	95,00	95,00	95,24	76,29	77,55	77,55
T15	91,18	92,59	92,59	92,59	77,50	62,50	62,50	62,50	83,78	74,63	74,63	74,63
T16	100	88,24	86,11	87,88	100	75,00	77,50	72,50	100	81,08	81,58	79,45
T17	100	90,00	87,80	94,74	100	90,00	90,00	90,00	100	90,00	88,89	92,31
T18	100	100	100	100	100	97,50	97,50	97,50	100	98,73	98,73	98,73
T19	100	78,72	81,82	76,00	100	92,50	90,00	95,00	100	85,06	85,71	84,44
T20	100	100	100	100	100	100	100	100	100	100	100	100
T21	91,67	78,95	79,49	79,49	82,50	75,00	77,50	77,50	86,84	76,92	78,48	78,48
T22	95,00	88,37	90,24	90,48	95,00	95,00	92,50	95,00	95,00	91,57	91,36	92,68
T23	84,09	78,57	78,57	80,49	92,50	82,50	82,50	82,50	88,10	80,49	80,49	81,48
T24	100	100	100	100	100	92,50	92,50	92,50	100	96,10	96,10	96,10
T25	100	100	97,56	97,56	100	100	100	100	100	100	98,77	98,77

ukuran citra asal. Dengan fungsi *kernel*-nya, NN mampu melakukan interpolasi terhadap nilai piksel citra baru dengan ukuran berbeda, sehingga memudahkan GNBC membedakan citra tekstur yang satu dengan yang lain. Secara visual, contoh hasil interpolasi dengan NN tersebut disajikan pada Tabel V.

Tingkat kinerja masing-masing metode pada Skenario 1 ini secara lebih mendalam dapat dilihat dari nilai presisi, sensitivitas, dan *F-score* pada Tabel VI. Hasil uji coba dengan *dataset* citra T01-T05 menunjukkan bahwa NN-GNBC secara umum memiliki presisi, sensitivitas, dan *F-score* (T01, T02, T03, T04, T05) jauh lebih baik daripada BL-GNBC, BC-GNBC, dan LZ-GNBC, walaupun untuk citra T03, semua metode menghasilkan nilai presisi yang sama. Semua hasil uji coba pada *dataset* tersebut menunjukkan bahwa kinerja NN-GNBC dalam mengklasifikasikan citra tekstur secara umum lebih baik dibandingkan dengan BL-GNBC, BC-GNBC, dan LZ-GNBC. BC-GNBC memperoleh presisi, sensitivitas, dan *F-score* relatif sama dengan LZ-GNBC, sedangkan BL-GNBC menghasilkan kinerja yang paling rendah di antara semuanya.

Untuk uji coba pada citra T06-T10, hasil menunjukkan bahwa selain menghasilkan akurasi terbaik, NN-GNBC juga menghasilkan presisi, sensitivitas, dan *F-score* (T06, T07, T08, T09, T10) relatif lebih tinggi dibandingkan BL-GNBC, BC-GNBC, dan LZ-GNBC, walaupun untuk citra T09, NN-GNBC memiliki sensitivitas lebih kecil dibandingkan BL-GNBC, BC-GNBC, dan LZ-GNBC. Maka, secara umum untuk percobaan pada citra T06-T10, NN-GNBC menghasilkan kinerja dalam klasifikasi citra tekstur lebih baik dibandingkan BL-GNBC, BC-GNBC, dan LZ-GNBC. Pada uji coba dengan *dataset* tersebut, BC-GNBC memperoleh presisi, sensitivitas, dan *F-score* relatif sama dengan LZ-GNBC, sedangkan BL-GNBC menghasilkan kinerja paling rendah.

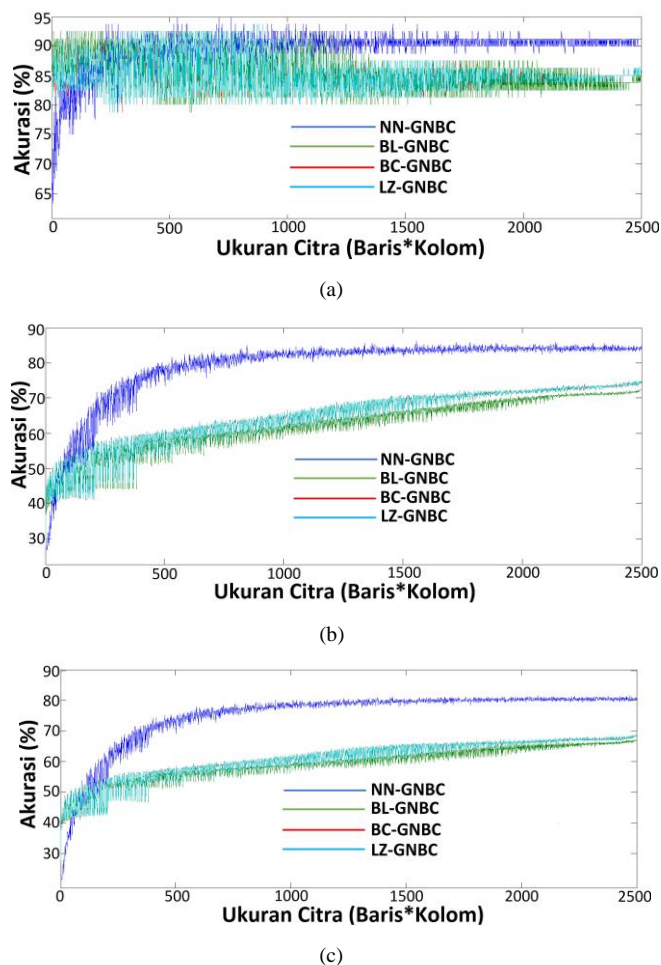
Untuk percobaan pada citra T11-T15, selain menghasilkan akurasi terbaik, secara umum NN-GNBC juga memiliki presisi, sensitivitas, dan *F-score* (T11, T12, T13, T14, T15) yang relatif

lebih baik dibandingkan BL-GNBC, BC-GNBC, dan LZ-GNBC, walaupun untuk T12 dan T15 presisinya lebih kecil dibanding yang lain. Oleh karena itu, secara umum uji coba NN-GNBC pada citra T11-T15 juga menghasilkan kinerja klasifikasi lebih baik daripada yang lainnya.

Percobaan dengan citra T16-T20 dan T21-T25 juga menunjukkan hasil yang sama. NN-GNBC memperoleh nilai presisi, sensitivitas, dan *F-score* (T16, T17, T18, T19, T20) relatif lebih tinggi dibandingkan dengan BL-GNBC, BC-GNBC, dan LZ-GNBC. Berdasarkan hasil uji coba pada *dataset* tersebut, NN-GNBC juga telah menghasilkan kinerja yang lebih unggul dibandingkan BL-GNBC, BC-GNBC, dan LZ-GNBC dalam mengklasifikasikan citra tekstur.

## B. SKENARIO 2

Uji coba pada Skenario 2 digunakan untuk mengetahui konsistensi kinerja metode yang diusulkan terhadap berbagai ukuran sampel *dataset* dan kelas/label yang digunakan. Untuk uji coba menggunakan citra tekstur T01-T02 (dua kelas), proses optimalisasi untuk memperoleh ukuran citra tekstur terbaik ditunjukkan pada Gambar 3. Dari Gambar 3(a), tampak bahwa pada ukuran citra (baris × kolom) < 518, NN-GNBC secara umum menghasilkan akurasi klasifikasi relatif lebih rendah dibandingkan metode yang lain. Akan tetapi, untuk ukuran citra tekstur (baris × kolom) ≥ 518, NN-GNBC memberikan nilai akurasi paling baik dibandingkan BL-GNBC, BC-GNBC, dan LZ-GNBC. Proses optimalisasi pada citra T01-T13 yang mewakili kelas sedang (13 kelas) dan citra T01-T025 yang mewakili kelas banyak (25 kelas) ditunjukkan pada Gambar 3. Dari Gambar 3(b), tampak bahwa dengan *dataset* T01-T13, NN-GNBC pada ukuran citra (baris × kolom) < 25 secara umum juga menghasilkan nilai akurasi relatif lebih rendah daripada metode yang lain, tetapi pada ukuran citra selain itu menghasilkan nilai akurasi paling baik. BC-GNBC dan LZ-GNBC memberikan hasil relatif sama, sedangkan BL-



**Gambar 3.** Proses optimalisasi untuk mendapatkan ukuran citra tekstur terbaik terhadap GNBC pada skenario 2, (a) T01-T02, (b) T01-T13, (c) T01-T25.

GNBC memberikan akurasi relatif lebih rendah dari semua metode yang diujikan. Pada *dataset* T01-T025, NN-GNBC pada ukuran citra (baris  $\times$  kolom)  $\geq 31$  memperoleh nilai akurasi yang paling baik dibandingkan metode yang lain, seperti ditunjukkan pada Gambar 3(c). Sementara itu, uji coba dengan citra tekstur T01-T09 (sembilan kelas), T01-T17 (17 kelas), dan T01-T21 (21 kelas) memberikan hasil relatif sama dengan uji coba pada *dataset* lainnya.

Proses optimalisasi pada *dataset* Skenario 2, yaitu T01-T02, T01-T03, T01-T05, T01-T07, T01-T09, T01-T11, T01-T13, T01-T17, T01-T21, dan T01-T25, menghasilkan ukuran citra pelatihan terbaik dan nilai akurasi klasifikasi terbaik seperti ditunjukkan pada Tabel III dan Tabel IV. Hasil uji coba tersebut menunjukkan bahwa secara keseluruhan NN-GNBC menghasilkan akurasi klasifikasi citra tekstur lebih baik dibandingkan dengan metode lainnya. Ukuran citra terbaik berbeda-beda untuk setiap *dataset* citra pelatihan. Sebagai contoh, untuk percobaan pada citra tekstur T01-T02, hasil optimalisasi menunjukkan bahwa NN-GNBC optimal (terbaik) pada ukuran citra  $9 \times 17$  (153) dengan akurasi 95%, sedangkan BL-GNBC optimal pada ukuran  $2 \times 2$  (4) dengan akurasi 92,5%, BC-GNBC pada ukuran  $17 \times 4$  (68) dengan akurasi 92,5%, serta LZ-GNBC pada ukuran  $44 \times 6$  (264) dengan akurasi 93,75%. Pada uji coba tersebut, metode NN-GNBC merupakan yang terbaik dari sisi akurasi, meskipun tidak pada ukuran citra. Sementara itu, untuk percobaan pada semua *dataset* citra, NN-GNBC mendapatkan hasil yang terbaik, baik dari sisi akurasi maupun ukuran citranya. Kendati untuk percobaan T01-T03, NN-GNBC bukan yang paling baik dari

sisi ukuran citra terbaiknya, nilai akurasi yang dihasilkan masih merupakan yang paling baik. Untuk membandingkan metode ekstraksi fitur lain, yaitu metode PCA, hasil uji coba menunjukkan bahwa ekstraksi fitur dengan PCA (NN-PCA-GNBC) memperoleh akurasi lebih buruk daripada metode lainnya pada semua *dataset* citra.

Apabila uji coba pada *dataset* citra T01-T02 (dua kelas), T01-T03 (tiga kelas), dan T01-T05 (lima kelas) mewakili uji coba dengan sedikit kelas, lalu T01-T07 (tujuh kelas), T01-T09 (sembilan kelas), T01-T11 (sebelas kelas), dan T01-T13 (13 kelas) mewakili uji coba dengan jumlah kelas sedang, dan T01-T17 (17 kelas), T01-T21 (21 kelas), dan T01-T25 (25 kelas) mewakili uji coba dengan banyak kelas, maka hasil uji coba secara keseluruhan menunjukkan bahwa NN-GNBC lebih konsisten (stabil) dengan tingkat akurasi klasifikasi citra tekstur paling baik. Selain itu, hasil ukuran citra pelatihan dengan NN-GNBC relatif lebih baik daripada metode yang lain.

Terlepas dari keberhasilan NN-GNBC memperoleh hasil terbaik dalam pengubahan ukuran citra tekstur pada skenario yang dilakukan, disadari bahwa ada beberapa keterbatasan dalam uji coba ini, di antaranya perlunya keterlibatan beberapa metode ekstraksi fitur dari objek citra dan beberapa metode klasifikasi dalam percobaan, sehingga dihasilkan kesimpulan yang lebih umum.

## V. KESIMPULAN

Optimalisasi metode interpolasi NN memberikan kontribusi yang signifikan untuk mendapatkan ukuran citra tekstur terbaik terhadap GNBC. Dengan melakukan optimalisasi metode interpolasi NN, dapat diketahui tingkat kontribusi metode interpolasi NN untuk meningkatkan kinerja GNBC berdasarkan nilai akurasi, presisi, sensitivitas, dan *F-score* dalam mengklasifikasikan citra tekstur. Hasil uji coba dengan menggunakan beberapa skenario *dataset* menunjukkan bahwa pengubahan ukuran citra tekstur berdasarkan optimalisasi interpolasi NN menghasilkan nilai akurasi yang lebih baik daripada penggunaan interpolasi *bilinear*, *bicubic*, dan Lanczos serta lebih baik daripada PCA. Di sisi lain, optimalisasi interpolasi citra tekstur menggunakan *bicubic* dan Lanczos menghasilkan kinerja relatif sama dan lebih baik daripada interpolasi *bilinear*. Hasil uji coba dengan beberapa *dataset* citra tekstur yang dianggap mewakili ukuran kelas kecil, sedang, dan besar juga menunjukkan bahwa metode interpolasi NN lebih konsisten memberikan hasil citra tekstur terbaik dengan tingkat akurasi klasifikasi GNBC lebih baik daripada metode interpolasi yang lain. Penentuan ukuran citra terbaik hasil optimalisasi menggunakan metode interpolasi diharapkan dapat menjadi alternatif untuk reduksi citra dan ekstraksi fitur dari objek citra sebelum dilakukannya proses klasifikasi.

## KONFLIK KEPENTINGAN

Penulis menyatakan bahwa tidak terdapat konflik kepentingan pada artikel berjudul “Citra Tekstur Terbaik untuk *Gaussian Naïve Bayes* dengan Interpolasi *Nearest Neighbor*”.

## KONTRIBUSI PENULIS

Konseptualisasi, Irwan Budi Santoso dan Shoffin Nahwa Utama; metodologi, Irwan Budi Santoso; perangkat lunak, Irwan Budi Santoso; validasi, Irwan Budi Santoso, Shoffin Nahwa Utama, dan Supriyono; analisis formal, Irwan Budi Santoso; investigasi, Irwan Budi Santoso; kurasi data, Irwan Budi Santoso; penulisan—penyusunan draf asli, Irwan Budi Santoso; penulisan—peninjauan dan penyuntingan, Shoffin Nahwa Utama dan Supriyono; visualisasi, Irwan Budi Santoso.

## UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada tim kelompok keilmuan prodi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang Indonesia yang telah mendukung terlaksananya penelitian ini.

## REFERENSI

- [1] M.S. Nixon dan A.S. Aguado, *Feature Extraction and Image Processing*, 1st ed., Oxford, Inggris: Newnes, 2002.
- [2] Y.-H. Shin, M.-J. Park, O.-Y. Lee, dan J.-O. Kim, "Deep orthogonal transform feature for image denoising," *IEEE Access*, vol. 8, hal. 66898–66909, Apr. 2020, doi: 10.1109/ACCESS.2020.2986827.
- [3] I.T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York, AS: Springer-Verlag, 2002.
- [4] S.H. Mahajan dan V.K. Harpale, "Adaptive and non-adaptive image interpolation techniques," *2015 Int. Conf. Comput. Commun. Control Automat.*, 2015, hal. 772–775, doi: 10.1109/ICCUBEA.2015.154.
- [5] G. Ramesh dan T.A. Prasath, "An aphoristic study on different interpolation techniques for medical image scaling and its comparative analysis," *2021 Int. Conf. Comput. Commun. Inform. (ICCCI)*, 2021, hal. 1–4, doi: 10.1109/ICCCI50826.2021.9402675.
- [6] D. újica-Vargas, Y. Mújica-Vargas, M.M. Cruz, dan A. Rendón-Castro, "Improvement of MRI images through heterogeneous interpolation techniques," *2019 Int. Conf. Electron. Commun. Comput. (CONIELECOMP)*, 2019, hal. 112–117, doi: 10.1109/CONIELECOMP.2019.8673230.
- [7] P. Bhatt, S. Patel, dan R. Pandit, "Comparative analysis of interpolation and texture synthesis method for enhancing image," *Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 2, no.1, hal. 278–283, Jan. 2013.
- [8] S. Safinaz dan A.V.R. Kumar, "VLSI realization of Lanczos interpolation for a generic video scaling algorithm," *2017 Int. Conf. Recent Adv. Electron. Commun. Technol. (ICRAECT)*, 2017, hal. 17–23, doi: 10.1109/ICRAECT.2017.37.
- [9] I.B. Santoso, Supriyono, C. Crysdian, dan K.F.H. Holle, "Optimization of naïve Bayes classifier to classify green open space object based on Google Earth image," *2018 Int. Seminar Res. Inf. Technol. Intell. Syst. (ISRITI)*, 2018, hal. 465–469, doi: 10.1109/ISRITI.2018.8864279.
- [10] P. Domingos dan M. Pazzani, "On the optimality of the simple Bayesian classifier under zero-one loss," *Mach. Learn.*, vol. 29, hal. 103–130, Nov. 1997, doi: 10.1023/A:1007413511361.
- [11] G.N. Norén dan R. Orre, "Case based imprecision estimates for Bayes classifiers with the Bayesian bootstrap," *Mach. Learn.*, vol. 58, hal. 79–94, Jan. 2005, doi: 10.1007/s10994-005-5010-y.
- [12] M. Ekdahl dan T. Koski, "Bounds for the loss in probability of correct classification under model based approximation," *J. Mach. Learn. Res.*, vol. 7, hal. 2449–2480, Nov. 2006.
- [13] M. Hall, "A decision tree-based attribute weighting filter for naïve Bayes," *Knowl.-Based Syst.*, vol. 20, no. 2, hal. 120–126, Mar. 2007, doi: 10.1016/j.knosys.2006.11.008.
- [14] T.-T. Wong, "Alternative prior assumptions for improving the performance of naïve Bayesian classifiers," *Data Min. Knowl. Discov.*, vol. 18, no. 2, hal. 183–213, Apr. 2009, doi: 10.1007/s10618-008-0101-6.
- [15] G. Shobha and S. Rangaswamy, "Machine learning," dalam *Handbook of Statistics 48: Deep Learning*, V. Gavindaraju, A.S.R.S. Rao, dan C.R. Rao, Eds., Cambridge, AS: Academic Press Publications, 2018, hal. 197–228, doi: 10.1016/bs.host.2018.07.004.
- [16] X. Wu *et al.*, "Top 10 algorithms in data mining," *Knowl. Inf. Syst.*, vol. 14, no. 1, hal. 1–37, Jan. 2008, doi: 10.1007/s10115-007-0114-2.
- [17] A.R. Webb dan K.D. Copesey, *Statistical Pattern Recognition*, 3rd ed., Hoboken, AS: John Wiley & Sons, Ltd., 2011.
- [18] R.C. Gonzalez, R.E. Woods, dan S.L. Eddins, *Digital Image Processing Using MATLAB*, 2nd ed., Knoxville, AS: Gatesmark Publishing, 2009.
- [19] S. Iqbal dkk., "Prostate cancer detection using deep learning and traditional techniques," *IEEE Access*, vol. 9, hal. 27085–27100, Feb. 2021, doi: 10.1109/ACCESS.2021.3057654.
- [20] S. Lazebnik, C. Schmid, dan J. Ponce (2003) The texture database, CVonline image database. [Online], [http://www-cvr.ai.uiuc.edu/ponce\\_grp/data/#texture](http://www-cvr.ai.uiuc.edu/ponce_grp/data/#texture), tanggal akses: 18-Apr-2017
- [21] D.M.W. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," 2020, *arXiv.2010.16061*.
- [22] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, hal. 861–874, Jun. 2006, doi: 10.1016/j.patrec.2005.10.010.
- [23] W. Zhao, Y. Lv, Q. Liu, dan B. Qin, "Detail-preserving image denoising via adaptive clustering and progressive PCA thresholding," *IEEE Access*, vol. 6, hal. 6303–6315, Des. 2017, doi: 10.1109/ACCESS.2017.2780985.
- [24] T. Acharya dan P.-S. Tsai, "Computational foundations of image interpolation algorithms," *Ubiquity*, vol. 2007, no. October, hal. 1–17, Okt. 2007, doi: 10.1145/1322464.1317488.
- [25] Z. Xingyu, W. Yong dan L. Xiaofei, "Approach for ISAR imaging of near-field targets based on coordinate conversion and image interpolation," *J. Syst. Eng. Electron.*, vol. 32, no. 2, hal. 425–436, Apr. 2021, doi: 10.23919/JSEE.2021.000036.
- [26] K.-L. Chung, C.-Y. Huang, dan C.-W. Kao, "An effective bicubic convolution interpolation-based iterative luma optimization for enhancing quality in chroma subsampling," *IEEE Access*, vol. 9, hal. 149744–149755, Nov. 2021, doi: 10.1109/ACCESS.2021.3125713.
- [27] C.-S. Tsai, H.-H. Liu, dan M.-C. Tsai, "Design of a scan converter using the cubic convolution interpolation with Canny edge detection," *2011 Int. Conf. Elect. Inf. Control Eng.*, 2011, hal. 5813–5816, doi: 10.1109/ICEICE.2011.5777979.
- [28] R.V. Sharan dan T.J. Moir, "Time-frequency image resizing using interpolation for scoustic event recognition with convolutional neural networks," *2019 IEEE Int. Conf. Signals Syst. (ICSigSys)*, 2019, hal. 8–11, doi: 10.1109/ICSIGSYS.2019.8811088.
- [29] A. Puziy, I. Gavrilov, K. Nosirov, dan A. Akhmedova, "Efficiency estimation of image resizing based on interpolating transformations," *2019 Int. Conf. Inf. Sci. Commun. Technol. (ICISCT)*, 2019, hal. 1–5, doi: 10.1109/ICISCT47635.2019.9011885.