BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Jurnal penelitian pendahulu yang pertama membahas tentang metode GLCM. Dalam jurnal yang berjudul "Identifikasi Tanda Tangan dengan Ekstraksi Fitur GLCM dan LBP" membahas tentang perbandingan kedua metode ekstraksi fitur yaitu metode GLCM dan metode LBP. Data tanda tangan yang digunakan dengan tinta hitam dari 15 orang, dimana masing-masing terdapat 10 tanda tangan. Data tanda tangan bertinta warna selain hitam dari 5 orang, masing-masing dengan 10 tanda tangan. Untuk data pelatihan diambil 7 tanda tangan dari masing-masing orang sehingga data latih berjumlah 105 tanda tangan untuk bertinta hitam dan 45 tanda tangan untuk bertinta warna selain hitam. Sedangkan data pengujian diambil 3 tanda tangan dari masing-masing orang sehingga data uji berjumlah 45 tanda tangan bertinta hitam dan 15 tanda tangan bertinta warna selain hitam. Dari hasil pengolahan citra didapatkan prosentase menggunakan ekstraksi fitur GLCM lebih besar dibandingkan prosentase menggunakan ekstraksi fitur LBP, yaitu GLCM mencapai 86,67% dan LBP 80,00% dengan tanda tangan bertinta hitam. Sedangkan hasil akurasi jika ditambahkan data tanda tangan bertinta warna selain hitam, untuk GLCM sebesar 80,00% dan LBP 78,33%. Sehingga dapat disimpulkan bahwa metode ekstralksi fitur dengan GLCM lebih baik dibandingkan dengan metode LBP.

Pada jurnal penelitian pendahulu membahas tentang metode klasifikasi K-NN. Jurnal yang berjudul "Algoritma K-Nearest Neighbor Classification Sebagai

Sistem Prediksi Predikat Prestasi Mahasiswa" yang membahas tentang akurasi metode K-NN dalam hal klasifikasi. Predikat prestasi mahasiswa diperoleh dari hasil sebuah prediksi. Proses prediksi dilakukan dengan menggunakan metode K-Nearest Neighbor (K-NN). Atribut yang digunakan dalam proses prediksi adalah Jenis Kelamin, Jenis Tinggal, Umur, Jumlah Satuan Kredit Semester (SKS), dan Jumlah Nilai Mutu (NM), sehingga dengan menerapkan algortima K-NN dapat dilakukan sebuah prediksi berdasarkan kedekatan dari histori data lama (training) dengan data baru (testing). Penentuan atribut ini berdasarkan hasil penelitian terdahulu yang memiliki kesamaan dalam kasus prediksi mahasiswa yang selanjutnya divalidasi oleh bagian Akademik Fakultas Sains dan Teknologi.Proses prediksi dilakukan terhadap Mahasiswa Program Studi Sistem Informasi angkatan 2014/2015 sebagai data testing dengan jumlah 50 data, serta berdasarkan dari data angkatan 2012/2013 sebagai data training dengan jumlah 165 data yang menghasilkan pengujian akurasi sebesar 82%. Hasil dari perhitungan algoritma K-NN diimplememetasikan terhadap sebuah Early Warning System (EWS).Output dari sistem yang dibangun dapat dijadikan sebagai acuan bagi Mahasiswa untuk meningkatkan prestasi dan predikat perkuliahan dimasa yang akan datang. Maka metode K-NN dinyatakan cukup akurat dalam klasifikasi.

2.2 Landasan Teori

2.2.1 Klasifikasi

Klasifikasi adalah proses menentukan suatu obyek kedalam suatu kelas atau kategori yang telah ditentukan (Raharjo, dkk., 2014). Komponen-komponen utama dari proses klasifikasi antara lain :

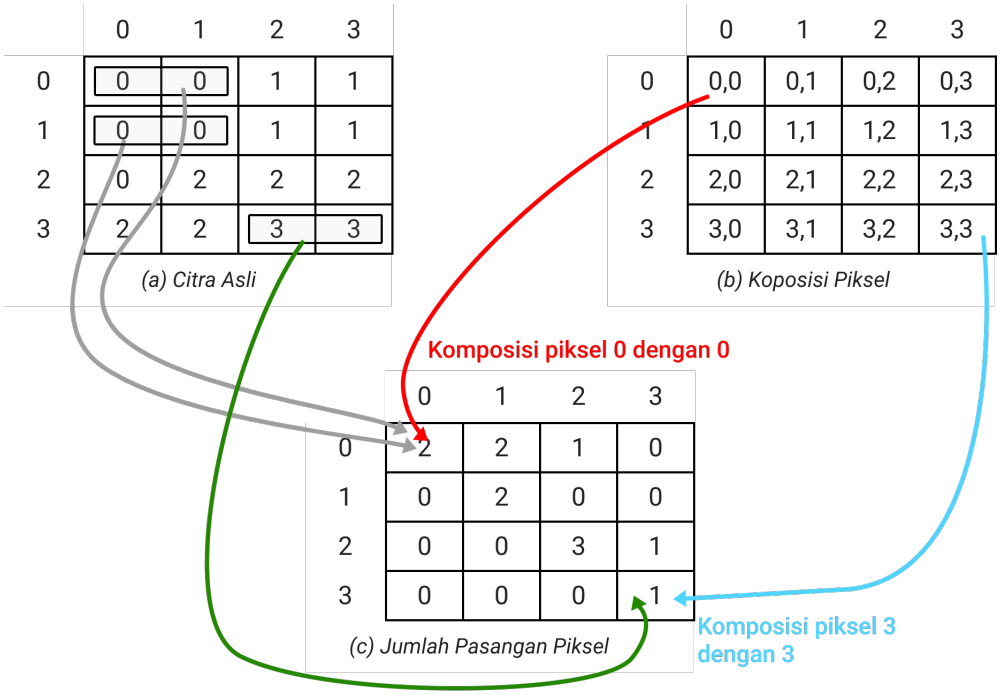
a. Kelas, merupakan variabel tidak bebas yang merupakan label dari hasil klasifikasi.
b. Predictor, merupakan variabel bebas suatu model berdasarkan dari karakteristik atribut data yang diklasifikasi.
c. Set data pelatihan, merupakan sekumpulan data lengkap yang berisi kelas dan predictor untuk dilatih agar model dapat mengelompokan ke dalam kelas yang tepat.
d. Set data uji, berisi data-data baru yang akan dikelompokan oleh model guna mengetahui akurasi dari model yang telah dibuat. (Widodo, dkk., 2013).

Klasifikasi citra merupakan salah satu teknik menginterpretasi citra digital. Klasifikasi citra didasarkan pada sampel masing-masing kelas, kualitasnya harus diperiksa dan dikuantifikasi setelahnya, dilakukan dengan pendekatan sampling dan hasil klasifikasi dengan kelas sesungguhnya dibandingkan. Perbandingan tersebut dilakukan dengan membuat error matrix yang dapat menghasilkan akurasi yang berbeda.

2.2.2 GLCM

Gray Level Co-occurrence Matrix (GLCM) adalah suatu metode yang digunakan untuk analisis tekstur/ekstraksi fitur. GLCM merupakan suatu matriks yang mengGambarkan frekuensi munculnya pasangan dua piksel dengan intensitas tertentu dalam jarak dan arah tertentu dalam citra (Prasetyo, 2011). Koordinat pasangan piksel memiliki jarak d dan orientasi sudut ϴ. Jarak direpresentasikan dalam piksel dan sudut direpresentasikan dalam derajat.

Kemudian dilakukan normalisasi terhadap matrik dengan menghitung probabilitas setiap elemen matrik(Pitiadani Br, 2017).



Gambar 2. 1 2 Piksel Matriks GLCM

Pada matriks Gambar 2.1 adalah *matrix framework*, yang perlu diolah menjadi matrik yang simetris dengan cara ditambahkan dengan hasil transposnya, terdapat pada Persamaan 2.1

$$\begin{bmatrix} 2 & 2 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 2 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 0 \\ 0 & 0 & 1 & 2 \end{bmatrix} \quad (2.1)$$

$\longrightarrow$ Transpos

GCLM sebelum dinormalisasi

Kemudian untuk menghapus ketergantungan pada citra, nilai pada GLCM harus di normalisasikan seperti pada Persamaan 2.2

$$\begin{bmatrix} \frac{4}{24} & \frac{2}{24} & \frac{1}{24} & \frac{0}{24} \\ \frac{2}{24} & \frac{4}{24} & \frac{0}{24} & \frac{0}{24} \\ \frac{1}{24} & \frac{0}{24} & \frac{6}{24} & \frac{1}{24} \\ \frac{0}{24} & \frac{0}{24} & \frac{1}{24} & \frac{2}{24} \end{bmatrix} \quad (2.2)$$

Untuk orientasi sudut terbentuk berdasarkan empat arah sudut yaitu, 0°, 45°, 90° dan 135°, dan jarak antar piksel sebesar 1 piksel (Surya, et al., 2017). Berikut adalah parameter ekstraksi fitur yang digunakan :

a. Kontras

Kontras merupakan hasil perhitungan yang berkaitan dengan jumlah keberagaman intensitas keabuan dalam citra.

$$Contrast = \sum_i \sum_j (i-j)^2 \, p(i,j) \quad (2.3)$$

Keterangan :

1. i : baris
2. j : kolom

b. Homogenitas

Homogenitas merupakan representasi dari ukuran nilai kesamaan variasi dari intensitas citra. Apabila semua nilai piksel memiliki nilai yang seragam maka homogenitas memiliki nilai maksimum.

$$Homogeneity = \sum_i \sum_j \frac{p(i,j)}{1+|i,j|} \quad (2.4)$$

c. Energi

Energi merupakan hasil perhitungan yang berkaitan dengan jumlah keberagaman intensitas keabuan dalam citra.

$$Energy = \sum_{i} \sum_{j} p^2(i,j) \tag{2.5}$$

d. Korelasi

Korelasi merupakan representasi dari keterkaitan linear pada derajat citragrayscale. Correlation berkisar dari -1 hingga 1.

$$Correlation = \sum_{i} \sum_{j} \frac{(i - \mu i)(j - \mu j)p(i,j)}{\sigma i \, \sigma j} \tag{2.6}$$

2.2.3 K-NN

K-Nearest Neighbor termasuk kelompok instance-based learning. Algoritma ini juga merupakan salah satu teknik lazy learning. K-NN dilakukan dengan mencari kelompok k objek dalam data training yang paling dekat (mirip) dengan objek pada data baru atau data testing (Leidiyana, 2013). Jika sebuah data query yang labelnya tidak diketahui diinputkan, maka K-Nearest Neighbor akan mencari k buah data learning yang jaraknya paling dekat dengan data query dalam ruang dimensi. Jarak antara data query dengan data learning dihitung dengan cara mengukur jarak antara titik yang merepresentasikan data query dengan semua titik yang merepresentasikan data learning dengan rumus Euclidian Distance. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan jarak Euclidean dengan rumus seperti pada persaman (2.5) berikut (T, Asahar Johar, 2017).

$$distance = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (2.7)$$

Keterangan :

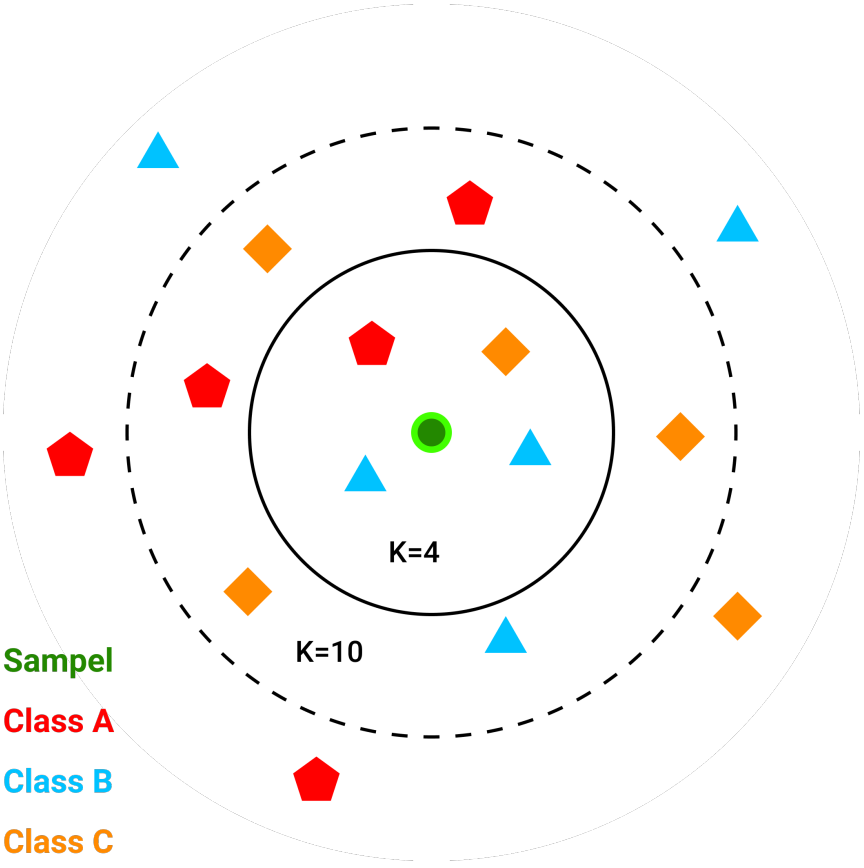
a. $distance$: Jarak Kedekatan

b. x : Data *Training*.

c. y : Data *Testing*.

d. i : Atribut individu antara 1 sampai dengan n.

e. n : jumlah atribut individu antara 1 sampai dengan n.



Gambar 2. 2 Contoh Alur Kerja K-NN

Pada Gambar 2.4 merupakan suatu contoh objek yamg berjumlah 15, pada objek tersebut terdapat 3 objek yang berbeda yaitu merah, biru dan kuning, untuk

objek yang berwarna merah berjumlah 5, untuk objek yang berwarna biru berjumlah 5 dan untuk objek berwarna kuning terdapat 5, kemudian untuk yang berwarna hijau adalah sebuah objek test. Pada lingkaran pertama terdapat 4 objek yang berbeda, diGambar tersebut telah di tetapkan bahwa nilai ketetanggaan atau nilai K adalah 4, dan pada lingkaran kedua terdapat 10 objek yang berarti ketetanggaannya adalah 10 atau K=10.

Pada objek hijau akan melakukan pengecekan objek yang ada terdekatnya hingga nilai K yang sudah di tetapkan, jika sudah menemukan objek terdekat sesuai nilai K, maka proses pengecekan objek selesai.

Sebagai ilustrasi dari penerapan algoritma KNN adalah misalnya terdapat data hasil survei dengan kuesioner, untuk meminta pendapat orang pada pengujian dengan dua atribut (ketahanan asam dan kekuatan), untuk mengklasifikasikan apakah suatu kertas tisu berkualitas baik atau tidak(Teknomo K, 2010). Berikut ini adalah empat sampel data training.

Tabel 2. 1 Data Training

X1 = Asam Durabilitas (detik)	X2 = Kekuatan (Kg/m^2)	Klasifikasi
7	7	Buruk
7	4	Buruk
3	4	Baik
1	4	Baik

Sebagai kasus, misalnya saat ini pabrik kertas telah menghasilkan jaringan baru yang lulus uji laboratorium dengan X1=3 dan X2=7. Untuk menebak

klasifikasi jaringan baru ini maka dilakukan perhitungan dengan menggunakan algoritma KNN. Adapun langkah–langkah untuk menghitung K tetangga terdekat dengan algoritma KNN adalah sebagai berikut :

a. Tentukan parameter K (jumlah tetangga terdekat). Misalkan K = 3.
b. Hitung jarak antara permintaan (data testing) dan contoh-contoh latihan semua (data training). Data training yang akan dihitung kedekatannya mempunyai koordinat (3,7).

Tabel 2. 2 Perhitungan Jarak

X1 = Asam Durabilitas (detik)	X2 = Kekuatan (Kg/m^2)	Square Jarak ke Contoh permintaan(3,7)
7	7	$(7-3)^2 + (7-7)^2 = 16$
7	4	$(7-3)^2 + (4-7)^2 = 25$
3	4	$(3-3)^2 + (4-7)^2 = 9$
1	4	$(1-3)^2 + (4-7)^2 = 13$

c. Urutkan jarak dan menentukan tetangga terdekat berdasarkan jarak terdekat ke-K.

Tabel 2. 3 Urutan Peringkat Jarak Minimum

X1 = Asam Durabilitas (detik)	X2 = Kekuatan (Kg/m^2)	Square Jarak ke Contoh permintaan(3,7)	Peringkat Jarak Minimum
7	7	$(7-3)^2 + (7-7)^2 = 16$	3
7	4	$(7-3)^2 + (4-7)^2 = 25$	4
3	4	$(3-3)^2 + (4-7)^2 = 9$	1

1	4	$(1-3)^2 + (4-7)^2 = 13$	2

d. Kumpulkan kategori Y dari baris tetangga terdekat. Pada baris kedua kategori tetangga terdekat (Y) tidak dimasukkan karena data tersebut peringkatnya lebih dari 3 tetangga terdekat.

Tabel 2. 4 Kumpulan Kategori Y Tetangga Terdekat

X1 = Asam Durabilitas (detik)	X2 = Kekuatan (Kg/m^2)	Square Jarak ke Contoh permintaan(3,7)	Peringkat Jarak Minimum	Apakah termasuk dalam tetangga terdekat	Y = Kategori terdekat tetangga
7	7	$(7-3)^2 + (7-7)^2 = 16$	3	Ya	Baik
7	4	$(7-3)^2 + (4-7)^2 = 25$	4	Tidak	-
3	4	$(3-3)^2 + (4-7)^2 = 9$	1	Ya	Baik
1	4	$(1-3)^2 + (4-7)^2 = 13$	2	Ya	Baik

e. Gunakan mayoritas sederhana dari kategori tetangga terdekat sebagai nilai prediksi contoh query.

Dari Tabel – Tabel yang telah diGambarkan, diperoleh dua kertas tisu baruberkualitas baik dan satu kertas tisu baru berkualitas buruk. Karena tetangga terdekat yang didapat lebih banyak yang berkualitas baik, maka dapat

disimpulkan bahwa kertas tisu baru yang lulus uji laboratorium dengan X1=3 dan X2=7 adalah termasuk dalam kategoi baik.

2.2.4 *K-fold Cross Validation*

K-fold cross validation biasa disebut dengan estimasi rotasi, yaitu membagi data menjadi kumpulan bagian K dengan jumlah yang sama dan menentukan data *train* dan data *test* (data latih dan data uji) sebanyak K. Teknik ini digunakan untuk memprediksi sampel dan memperkirakan keakuratan sebuah sampel ketika dipraktikkan. Untuk perhitungannya yaitu melakukan perulangan, salah satu pada kumpulan akan dijadikan sebagai data pengujian dan kumpulan data K lainnya dijadikan sebagai data pelatihan ().

Proses pertama diawali dengan membagi data sejumlah nilai n-fold yang diinginkan dan akan digunakan untuk validasi data yang dibagi sesuai nilai n dengan ukuran yang sama. Selanjutnya proses latih dan proses uji dilakukan sebanyak K. Untuk penggunaan jumlah fold yang terabaik untuk uji validasitas adalah K=5 atau K=10 (Hastie et al, 2009). Berikut skema 5-fold Cross Validation pada Gambar 2.3

Gambar 2. 3 Urutan Untuk Menghitung Data Pengujian dan Data Pelatihan

Berikut cara kerja dari *K-fold Cross Validation* :

1. Total kumpulan sampel dibagi menjadi N bagian.
2. Fold pertama adalah bagian pertama untuk menjadi data uji (*Test*) dan sisanya akan dijadikan data latih (*Train*). Pada selanjutnya akan melakukan perhitungan akurasi, *error rate*, *Sensitivity*, dan *Specificity*. Berikut Persamaan dari perhitungannya tersebut.
3. Fold kedua adalah bagian kedua untuk menjadi data uji (*Test*) dan sisanya termasuk fold pertama akan dijadikan data latih (*Train*). Selanjutnya akan melakukan perhitungan sesuai data tersebut.
4. Perhitungannya akan seperti itu seterusnya hingga mencapai fold ke-n.

2.2.5 *Confusion Matrix*

Confusion matrix adalah sebuah Tabel yang menggmbarkan performa pada sebuah sampel secara spesifik. Setiap baris dari matrix terdapat kelas aktual dan setiap kolom dari matrix terdapat kelas prediksi, seperti pada Tabel 2. Pada Tabel 2.5 merupakan Tabel multiclass confusion matrix 2x2.

Tabel 2. 5 Model *Confussion Matrix*

		PREDIKSI	
		POSITIF	NEGATIF
AKTUAL	POSITIF	True Positif (TP)	False Negatif (FN)
	NEGATIF	False Positif (FP)	True Negatif (TN)

Keterangan :

1. True Positif : Jumlah data yang aktual dengan kelas positif dan sampel yang memprediksi positif
2. True Negatif : Jumlah data yang aktual dengan kelas negatif dan sampel yang memprediksi negatif
3. False Positif : Jumlah data yang aktual dengan kelas negatif, tapi sampel memprediksi positif
4. False Negatif : Jumlah data yang aktual dengan kelas positif, tapi sampel memprediksi negatif

Dengan menggunakan 4 data tersebut metode ini dapat digunakan untuk melakukan perhitungan atau mengukur performa pada sebuah sampel, *Accuracy*, *erro rate*, *Precission, Sensitivity (Recall)*, *Specificity,* dan *F1 Score*.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.8}$$

Keterangan :

Accuracy : untuk menghitung seberapa akurat sampel dalam mengklasifikasikan data dengan benar.

$$Error\ Rate\ = \frac{FP + FN}{TP + TN + FP + FN} \tag{2.9}$$

Keterangan :

Error Rate : untuk menghitung seberapa akurat sampel dalam mengklasifikasikan data yang salah atau tidak sesuai.

$$Sensitivity = \frac{TP}{TP + FN} \tag{2.10}$$

Keterangan :

Sensitivity / Recall : untuk mengukur seberapa tepat sampel yang di prediksi benar positif di bandingkan dengan seluruh data yang benar positif.

$$Specificity = \frac{TN}{TN + FP} \tag{2.11}$$

Keterangan :

Specificity : untuk mengukur kebenaran sebuah sampel yang di prediksi negatif di bandingkan dengan seluruh data yang negatif