BAB III
LANDASAN TEORI

3.1 Tahap Praproses

Sebelum ekstraksi fitur, dilakukan operasi *cropping* terlebih dulu sehingga fokus penelitian hanya pada bagian lesi. Proses *cropping* ini dilakukan oleh radiolog yang telah mengetahui posisi lesi. Hasil dari proses *cropping* tersebut digunakan sebagai citra masukkan untuk proses ekstraksi fitur.

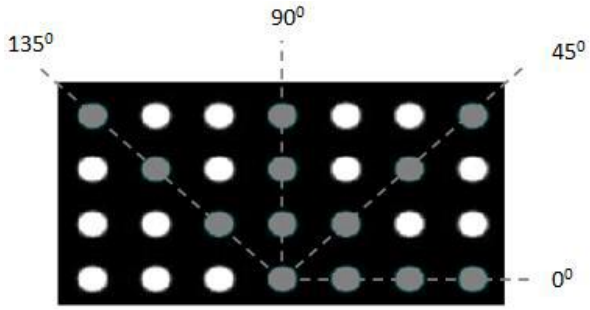
3.2 Tahap Ekstraksi Fitur

Citra hasil proses *cropping* digunakan sebagai masukkan dalam tahap ekstraksi fitur yang menggunakan GLCM.

Gray Level Co-occurrence Matrices (GLCM) merupakan salah satu metode yang dapat digunakan untuk melakukan ekstraksi fitur tekstur. GLCM menggunakan perhitungan tekstur pada orde dua. Pengukuran tekstur pada orde pertama menggunakan perhitungan statistika didasarkan pada nilai piksel citra asli, seperti varians, dan tidak memperhatikan hubungan ketetanggaan piksel. Berbeda dengan GLCM yang memperhitungkan hubungan antarpasangan dua piksel citra asli. Misalkan f(x,y) adalah citra dengan ukuran N_x dan N_y yang memiliki piksel dengan kemungkinan hingga level derajat keabuan (L level) dan $\vec{\boldsymbol{r}}$ adalah vector arah ofset spasial. $\boldsymbol{GLCM}_{\vec{r}}(\boldsymbol{i},\boldsymbol{j})$ didefinisikan sebagai jumlah piksel dengan $\boldsymbol{j} \in \boldsymbol{1},..,\boldsymbol{L}$ yang terjadi pada ofset $\vec{\boldsymbol{r}}$ terhadap piksel dengan nilai $\boldsymbol{i} \in \boldsymbol{1},..,\boldsymbol{L}$ yang dapat dinyatakan dengan persamaan (1) (Kadir & Adhi 2012) (Mediatrix 2015) sebagai berikut.

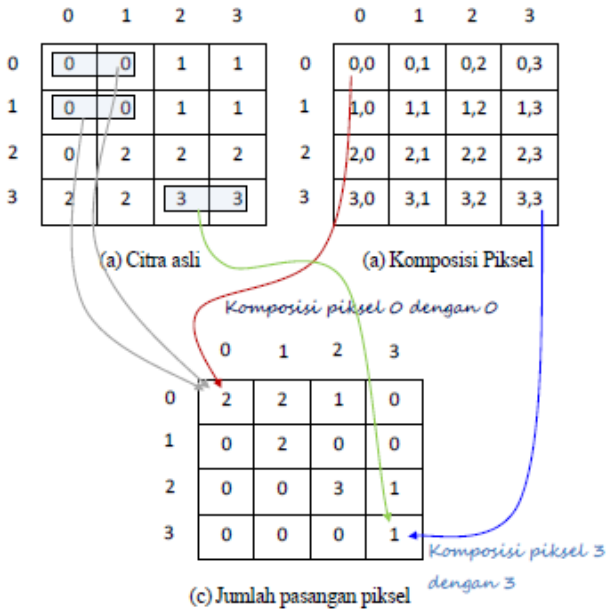
$$\begin{aligned}\boldsymbol{GLCM}_{\vec{r}}(\boldsymbol{i},\boldsymbol{j}) &= \#\{(\boldsymbol{x_1},\boldsymbol{y_1}),(\boldsymbol{x_2},\boldsymbol{y_2}) \in (\boldsymbol{N_x},\boldsymbol{N_y})\boldsymbol{X}(\boldsymbol{N_x},\boldsymbol{N_y})|\boldsymbol{f}(\boldsymbol{x_1},\boldsymbol{y_1}) \\ &= \boldsymbol{j}^{\vec{r}} = \overrightarrow{(\boldsymbol{x_2}-\boldsymbol{x_1},\boldsymbol{y_2}-\boldsymbol{y_1})}\}\end{aligned} \quad (1)$$

Dalam hal ini, # menunjukkan jumlah elemen dari himpunan. Ofset $\vec{r}$ dapat berupa sudut atau jarak. Gambar 3.1 memperlihatkan empat arah untuk GLCM (Kadir & Adhi 2012).



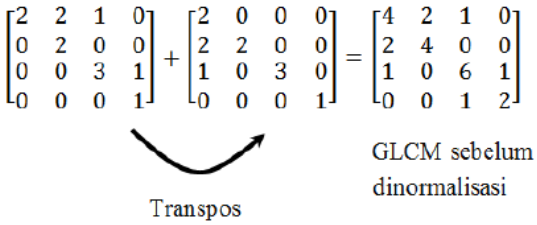
Gambar 3. 1 Contoh arah untuk GLCM dengan sudut 0^0, 45^0, 90^0 dan 135^0

Untuk ilustrasi, ketetanggan piksel dapat dipilih ke arah timur (kanan). Salah satu cara untuk merepresentasikan hubungan ini yaitu berupa (1,0), yang menyatakan hubungan dua piksel yang berjajar horizontal dengan piksel bernilai 1 diikuti dengan piksel bernilai 0. Berdasarkan komposisi tersebut, jumlah kelompok piksel yang memenuhi hubungan tersebut dihitung. Hal ini diiliustrasikan pada Gambar 3. 2 berikut.



Gambar 3. 2 Penentuan awal matriks GLCM berbasis pasangan dua piksel

Matriks pada Gambar 3.2 dinamakan *matrix framework*. Matriks ini perlu diolah menjadi matriks simetris dengan cara menambahkan dengan hasil transposnya, seperti pada Gambar 3.3 berikut (Kadir & Adhi 2012).



Gambar 3. 3 Contoh pembentukan matriks GLCM yang simetris

Untuk menghilangkan ketergantungan pada ukuran citra, nilai-nilai elemen GLCM perlu dinormalisasi sehingga jumlahnya bernilai 1. Dengan demikian, contoh di depan akan menjadi sebagai berikut (Kadir & Adhi 2012).

$$\begin{bmatrix} \frac{4}{24} & \frac{2}{24} & \frac{1}{24} & \frac{0}{24} \\ \frac{2}{24} & \frac{4}{24} & \frac{0}{24} & \frac{0}{24} \\ \frac{1}{24} & \frac{0}{24} & \frac{6}{24} & \frac{1}{24} \\ \frac{0}{24} & \frac{0}{24} & \frac{1}{24} & \frac{2}{24} \end{bmatrix}$$

Untuk mendapatkan fitur GLCM, hanya beberapa besaran yang digunakan yaitu *Angular Second Moment* (ASM) atau energi, kontras, korelasi dan homogenitas. ASM/energi merupakan ukuran homogenitas citra dan memberikan jumlah elemen yang dikuadratkan, dapat dihitung menggunakan persamaan(2) (Kadir & Adhi 2012) berikut.

$$\boldsymbol{ASM} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{L}} \sum_{\boldsymbol{j=1}}^{\boldsymbol{L}} \boldsymbol{GLCM}(\boldsymbol{i},\boldsymbol{j})^{\boldsymbol{2}} \qquad (2)$$

Dalam hal ini, L menyatakan jumlah level yang digunakan untuk komputasi.

Kontras merupakan ukuran keberadaan variasi aras keabuan piksel citra dihitung dengan menggunakan persamaan(3) berikut (Kadir & Adhi 2012).

$$\boldsymbol{Kontras} = \sum_{\boldsymbol{n=1}}^{\boldsymbol{L}} \boldsymbol{n^2} \{ \sum_{\boldsymbol{|i-j|=n}} \boldsymbol{GLCM(i,j)} \} \quad (3)$$

Korelasi merupakan ukuran ketergantungan linear antar nilai aras keabuan atau seberapa besar hubungan antara satu piksel dengan piksel tetangganya dalam citra dihitung menggunakan persamaan(4) berikut (Kadir & Adhi 2012).

$$\boldsymbol{Korelasi} = \frac{\sum_{\boldsymbol{i=1}}^{\boldsymbol{L}} \sum_{\boldsymbol{j=1}}^{\boldsymbol{L}} \boldsymbol{(i,j)(GLCM(i,j) - \mu_i' \mu_j')}}{\boldsymbol{\sigma_i' \sigma_j'}} \quad (4)$$

Keterangan :

$\mu_i' = \sum_{i=1}^{L} \sum_{j=1}^{L} i * GLCM(i,j)$

$\mu_{ji}' = \sum_{i=1}^{L} \sum_{j=1}^{L} j * GLCM(i,j)$

$\sigma_j^2 = \sum_{i=1}^{L} \sum_{j=1}^{L} GLCM(i,j)(i - \mu_i')^2$

$\sigma_i^2 = \sum_{i=1}^{L} \sum_{j=1}^{L} GLCM(i,j)(i - \mu_i')^2$

Homogenitas atau *Inverse Difference Moment* merupakan ukuran kedekatan distribusi elemen dalam GLCM, dihitung menggunakan persamaan(5) berikut (Kadir & Adhi 2012).

$$\boldsymbol{IDM} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{L}} \sum_{\boldsymbol{j=1}}^{\boldsymbol{L}} \frac{\boldsymbol{GLCM(i,j)^2}}{\boldsymbol{1 + (i-j)^2}} \quad (5)$$

3.3 Tahap Klasifikasi

Secara sederhana, Naïve Bayes mengasumsikan bahwa nilai suatu fitur tidak berkaitan dengan keberadaan fitur lain, terhadap variabel kelas yang ditentukan. Naïve Bayes menganggap bahwa masing-masing fitur memiliki probabilitas kontribusi secara independen, terlepas dari ada atau tidaknya fitur lain. Meskipun desain Naïve Bayes tampak sederhana, metode ini bekerja cukup baik untuk menyelesaikan masalah-

masalah yang rumit. Untuk beberapa jenis model probabilitas, Naïve Bayes dapat dilatih dengan sangat efisien dalam *supervised learning* (pembelajaran terbimbing). Model probabilitas untuk klasifikasi adalah model kondisional yang bergantung pada variabel C dengan sejumlah kecil hasil atau kelas yang tergantung pada beberapa variabel fitur X_1 sampai X_n yang ditulis dalam persamaan(6) berikut (Zhou et al. 2015).

$$p(C|X_1 \ldots . X_{n)} \tag{6}$$

Namun, jika sebuah fitur memiliki nilai yang besar atau jika sejumlah fitur n besar, maka model tersebut tidak layak pada tabel probabilitas. Oleh karena itu, persamaan(6) diformulasi ulang seperti pada persamaan(7) berikut.

$$p(C|X_1, \ldots, X_n) = \frac{p(C)p(X_1, \ldots, X_n|C)}{p(X_1, \ldots, X_n)} \tag{7}$$

Dalam analisis Bayesian, klasifikasi akhir dihasilkan dengan menggabungkan kedua sumber informasi, sebelum kejadian dan kemungkinan kejadian, untuk membentuk probabilitas posterior menggunakan aturan Bayes, sehingga persamaan(7) dapat dirumuskan dalam persamaan(8) berikut.

$$Posterior = \frac{likelihood \times prior}{evidence} \tag{8}$$

Dalam prakteknya, kepentingan hanya terdapat di pembilang, karena penyebut tidak tergantung pada C dan nilai-nilai fitur X_i yang diberikan, sehingga penyebut bersifat konstan, jadi hanya perlu memperbesar nilai $p(C)p(X_1, \ldots, X_n|C)$.

Sekarang, asumsi kondisi independen "Naïve" bersyarat akan tampak. Asumsikan bahwa setiap fitur X_i adalah independen secara kondisional dari setiap fitur lainnya X_i untuk $j \neq i$, terhadap kategori C yang dirumuskan dalam persamaan(9) berikut.

$$p(X_1, \ldots, X_n|C) = \prod_{i=1}^{n} p(X_i|C \tag{9}$$

Dimana probabilitas $\mathrm{P}(\mathrm{X_1}|\mathrm{C}, \mathrm{P}(\mathrm{X_2}|\mathrm{C} \ldots \mathrm{P}(\mathrm{X_n}|\mathrm{C}$ dapat diperkirakan dengan sampel pelatihan. Berdasarkan perhitungan ini, dapat diperoleh probabilitas posterior dari sampel untuk masing-masing kelas. Selanjutnya, berdasarkan kriteria posterior

maksimum Bayesian, dapat dipilih kelas dengan probabilitas posterior terbesar sebagai label kelas.

3.4 Tahap *Feature Reduction*

Metode yang digunakan untuk reduksi data dibagi menjadi dua yaitu : Wrapper dan Filter. Model pendekatan Wrapper menggunakan metode klasifikasi itu sendiri untuk mengukur tingkat kepentingan dari sekumpulan fitur, selanjutnya fitur dipilih tergantung pada model pengklasifikasi yang digunakan. Metode Wrapper biasanya memberikan kinerja yang lebih baik daripada metode filter karena proses seleksi fitur dioptimalkan untuk algoritme klasifikasi. Akan tetapi, metode Wrapper terlalu mahal untuk data berdimensi besar karena kompleksitas komputasi dan waktu dikarenakan masing-masing fitur harus dievaluasi dengan algoritme klasifikasi. Pendekatan filter mendahului proses klasifikasi yang sebenarnya. Pendekatan ini bersifat independen dari algoritme pembelajaran, komputasi yang sederhana, cepat dan dapat terukur. Dengan menggunakan metode filter, proses seleksi fitur hanya dilakukan sekali dan kemudian dapat digunakan sebagai input untuk pengklasifikasi yang berbeda. Berbagai feature ranking dan seleksi fitur telah digunakan seperti *Correlation-based Feature Selection* (CFS), *Principal Component Analysis* (PCA), *Gain Ratio* (GR) *attribute evaluation*, *Chi-square feature evaluation, Fast Correlation-based Feature Selection* (FCBF), *Information gain*, *Euclidean distance*, *i-test*, *Markov blanket filter*. Beberapa dari metode filter ini tidak melakukan seleksi fitur tetapi hanya *feature ranking* dan oleh karena itu, metode tersebut dikombinasikan dengan metode pencarian untuk mengetahui jumlah atribut (Karegowda et al. 2010).

Metode Gain Ratio yang digunakan dalam penelitian ini berhubungan dengan metode information gain. Information gain digunakan untuk memilih atribut uji pada setiap node dari pohon keputusan dan cenderung memilih atribut yang memiliki sejumlah nilai yang besar. Variabel S terdiri dari sampel data s dengan m kelas yang berbeda. Informasi yang diharapkan, dibutuhkan untuk mengklasifikasi sampel yang dirumuskan dalam persamaan(10) berikut (Karegowda et al. 2010)

$$I(S) = -\sum_{i=1}^{m} p_i \, log_2(p_i) \tag{10}$$

dimana p_i adalah probabilitas sampel acak merupakan milik kelas C_i dan diperkirakan dengan s_i/s.

Atribut A memiliki nilai v yang berbeda dan s_{ij} merupakan jumlah sampel kelas C_i dalam S_j. S_j berisi sampel S yang memiliki nilai a_j dari A. Entropi atau informasi yang diharapkan berdasarkan partisi menjadi himpunan bagian A, yang dirumuskan dalam persamaan(11) berikut

$$E(A) = -\sum_{i=1}^{m} I(S) \frac{s_{1i} + s_{2i} + \cdots s_{mi}}{s} \tag{11}$$

Pengkodean informasi diperoleh dengan persamaan(12) berikut

$$Gain(A) = I(S) - E(A) \tag{12}$$

Gain ratio digunakan untuk normalisasi information gain menggunakan nilai yang dirumuskan dalam persamaan(13) berikut

$$SplitInfo_A(S) = -\sum_{i=1}^{v} (|S_i|I|S|)log_2(|S_i|I|S|) \tag{13}$$

Nilai pada persamaan(13) menunjukkan informasi dihasilkan dengan memisahkan data uji S menjadi bagian v sesuai dengan hasil v dari uji atribut A. Gain ratio dapat ditentukan dengan persamaan(14) berikut

$$Gain\ Ratio(A) = Gain(A)/SplitInfo_A(S) \tag{14}$$

Atribut yang memiliki gain ratio tertinggi dipilih sebagai *splitting attributes*.

3.5 Indeks Pengukuran

Pengukuran yang dilakukan dalam penelitian ini adalah untuk mengetahui tingkat kesuksesan terhadap proses-proses yang telah dilakukan. Pengukuran yang akan dilakukan yaitu pengukuran terhadap kinerja dari metode ekstraksi fitur maupun metode klasifikasi. Pengukuran dari proses klasifikasi ditentukan dengan nilai-nilai berikut :

1. Akurasi

Nilai akurasi dari hasil klasifikasi dapat diperoleh dengan menghitung jumlah klasifikasi yang benar dan sesuai target dibagi dengan jumlah klasifikasi yang berbeda dengan target dari semua kelas. Akurasi dirumuskan dalam persamaan(10)

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} X\ 100\% \quad (15)$$

dengan TP (*True Positive*) adalah jumlah data benar pada target yang terklasifikasi benar pada sistem, TN (*True Negative*) adalah jumlah data salah pada target yang terklasifikasi salah pada sistem, FP (*False Positive*) merupakan representasi jumlah data salah pada target yang terklasifikasi benar pada sistem dan FN (*False Negative*) merupakan representasi jumlah data benar pada target yang terklasifikasi salah pada sistem. Nilai-nilai tersebut akan tampil dalam bentuk *confusion matrix*.

2. Sensitivitas

Sensitivitas merupakan ukuran kemampuan sistem untuk melakukan prediksi terhadap data yang dianggap benar sesuai dengan TPR (*True Positive Rate*). Sensitivitas dapat dirumuskan dalam persamaan(11) berikut.

$$sensitivity = \frac{TP}{TP + FN} X\ 100\% \quad (16)$$

3. Spesifisitas

Spesifisitas berkebalikan dengan sensitivitas yaitu kemampuan sistem untuk melakukan prediksi terhadap data yang dianggap salah sesuai dengan TNR (*True Negative Rate*). Spesifisitas dapat dirumuskan dalam persamaan(12) berikut.

$$specificity = \frac{TN}{TN + FP} X\ 100\% \quad (17)$$