MATHunesa
Jurnal Ilmiah Matematika *Volume 9 No 1*
e-ISSN : 2716-506X | p-ISSN : 2301-9115 *Tahun 2021*

KLASIFIKASI K-NEAREST NEIGHBOR CHEST X-RAY PASIEN COVID-19 DENGAN HARALICK FEATURES DAN HISTOGRAM OF ORIENTED GRADIENT

Christian Adi Nugroho
Jurusan Matematika, FMIPA, Universitas Negeri Surabaya
e-mail : christian.17030214024@mhs.unesa.ac.id

Abstrak

Pandemi Covid-19 memiliki dampak serius pada kehidupan masyarakat. Salah satu langkah penting untuk mengatasi pandemi ini terletak pada kemampuan tenaga medis untuk mengidentifikasi pasien yang terinfeksi Covid-19 secara dini. Kemudian segera lakukan prosedur pengobatan dan isolasi pasien. Mendeteksi Covid-19 dari *radiograph* pasien mungkin menjadi salah satu cara tercepat untuk mengidentifikasi pasien Covid-19, yang didukung oleh penelitian sebelumnya yang menunjukkan gejala *abnormal* pada *radiograph* dada pasien Covid-19. Untuk mendeteksi pasien Covid-19 dari rontgen dada *(CXR)* yang terinspirasi dari penelitian sebelumnya yang menggunakan *Artificial Intelligence*, aplikasi *classifier Machine Learning k-Nearest Neighbor* telah dipelajari untuk hal yang sama. 1000 *CXR* diperoleh dari posisi *Anterior-Posterior* berlabel dari *dataset* COVID-Xray-5k, kemudian dipartisi dengan *random sampling,* 80% untuk *training set* dan sisanya untuk *test set*. Citra *CXR* yang ada dikonversi menjadi citra *grayscale* dimana diperoleh 149 fitur; 5 fitur adalah *Haralick Features* dan 144 fitur berasal dari *Histogram of Oriented Gradient*. Hasil klasifikasi dengan nilai estimasi k, dengan k = 10 mencapai akurasi rata-rata di atas 90% untuk jarak atau *metric* Euclid, Mahalanobis, Cosine, dan Cityblock. Oleh karena *CXR* pasien Covid-19 yang tersedia untuk umum terbatas, diperlukan penelitian terhadap dataset yang memiliki jumlah *CXR* pasien Covid-19 lebih banyak untuk menguji keakuratan *classifier*.
Kata kunci: *K-Nearest Neighbor, Covid-19, Haralick Features, Histogram of Oriented Gradient*

Abstract

Covid-19 pandemic has a severe impact on people's lives. One of the crucial steps to counter this pandemic lies within the ability of medical personnel to identify patients infected with Covid-19 early, is paramount. Then immediately carry out treatment procedures and patient isolation. Detecting Covid-19 from a radiograph may be one of the fastest ways to identify Covid-19 patients, supported by previous studies that showed abnormal symptoms on the chest radiograph of Covid-19 patients. To detect Covid-19 patients from chest x-rays (CXR), inspired by previous studies using Artificial Intelligence, the application of the Machine Learning classifier k-Nearest Neighbor were studied for the same thing. 1000 CXR were obtained from labelled Anterior-Posterior positions from the COVID-Xray-5k dataset, then partitioned 80% by random sampling for the training set and the rest for the test set. The existing image converted into a grayscale image where 149 features obtained; 5 features are Haralick Features and 144 features are from the Histogram of Oriented Gradient. The classification results with the estimated value of k, with k = 10 achieve accuracy above 90% for the distance or metric Euclid, Mahalanobis, Cosine, and Cityblock. CXR of Covid-19 patients which is publicly available is limited. Due to the reason, research on datasets that have a larger number of Covid-19 patient CXRs to test the accuracy of the classifier is required.
Keywords : *K-Nearest Neighbor, Covid-19, Haralick Features, Histogram of Oriented Gradient*

PENDAHULUAN

Mengingat terjadinya beberapa kali pandemi berskala *global outbreak* seperti *Spanish Flu*, *Avian Flu*, *Swine Flu*, *SARS*, *MERS*, dan pada Desember 2019, yaitu *outbreak* CoVid-19 (Liu, Kuo, & Shih, 2020) yang menginfeksi 90 juta orang dan merenggut nyawa hampir 2 juta jiwa dan masih mewabah hingga saat ini (World Health Organization Emergency Dashboard, 2021), menyebabkan tenaga medis membutuhkan cara-cara untuk mengidentifikasi penyakit yang diderita pasien dengan tepat sebelum memberikan tindak medis lanjutan untuk pasien yang diduga terinfeksi Covid-19.

Kecepatan untuk mendiagnosa pasien yang diduga terinfeksi adalah penting. Ini dikarenakan prosedur perawatan standar pasien Covid-19 mengharuskan prosedur isolasi agar dapat dilakukan segera demi mengurangi kemungkinan pasien menginfeksi orang yang masih sehat. Perlu diingat juga, bahwa dokumentasi dan penelitian mengenai Covid-19 di ranah medis sendiri masih terbilang baru.

Salah satu cara tenaga medis dapat mengidentifikasi Covid-19, selain menggunakan *RT-PCR (Real Time Reverse Transcription Polymerase Chain Reaction)* dan tes antigen, adalah dengan mengambil *chest x-ray (CXR)* atau *photo thorax/waters* sebagai

tindak observasi (Barry, Obadia, El Hajjam, & Carlier, 2020; Jacobi, Chung, Bernheim, & Eber, 2020; Ruttens et al., 2020). Opsi lainnya selain penggunaan sinar-x adalah dengan melakukan *CT-scan* atau *computer tomography* sebagai tindak observasi *non-invasive* (Saunders, 2008).

Keuntungan *x-ray imaging*, dibandingkan *CT* dan *MRI (magnetic resonance image)*, adalah murah (Qin, Yao, Shi, & Song, 2018), dan banyak tersedia di rumah sakit umumnya. *X-ray medical imaging* adalah yang paling umum dilakukan di seluruh dunia. Bahkan di Amerika sendiri, paling tidak, di satu rumah sakit, sepertiga rekam medisnya, baik *analog* maupun *digital*, adalah foto x–ray *thorax* atau *CXR* (Daffner & Hartman, 2013).

Dewasa ini, menggunakan input rekam medis pasien secara digital, proses identifikasi penyakit dapat dilakukan dengan bantuan *computer* atau alat komputasi menggunakan *Machine Learning (ML)* atau *Artificial Intelligence (AI)*. Ini membawa pada tujuan ditulisnya artikel, yaitu membuat *classifier* yang dapat mengidentifikasi pasien yang terinfeksi Covid-19 berdasarkan hasil *CXR* pasien tersebut yang sudah disimpan menjadi *file digital*.

Dataset yang digunakan adalah *open access dataset* dari penelitian sejenis terdahulu, yang terdiri dari kompilasi citra *CXR digital multiformat* seperti *file* '.jpeg', atau '.jpg' yang sudah dilabeli oleh ahli.

Pada kesempatan ini, akan dibuat *classifier ML non-parametric*, yaitu *K-Nearest Neighbor (k-NN)* dengan menggunakan *feature* atau *descriptor Gray-Level Co-Occurrence Matrices (GLCM)* atau sering disebut *Haralick Features* dan *Histogram of Oriented Gradient (HOG)* dari *CXR* pasien. Kedua *feature* ini sering digunakan dalam *medical image classification*; seperti klasifikasi tumor otak (S & Dharun, 2016; Zulpe & Pawar, 2012), polip (Iwahori, Hattori, Adachi, Bhuyan, & Robert, 2015), dan *mammogram* (Farhan & Kamil, 2020).

KAJIAN TEORI

CHEST X-RAY

Chest X-ray (CXR) atau *chest radiograph* pada dasarnya adalah foto Rontgen yang dilakukan pada daerah atas diafragma atau bagian dada. CXR diperoleh dengan menembakkan sinar-X pada bagian dada dan ditangkap (dicetak) pada *film* atau divisualisasikan sebagai citra *digital*, disimpan menjadi *file* citra *digital*, menggunakan komputer (Jaeger et al., 2014). Umumnya, *CXR* digunakan untuk mendiagnosa pelbagai penyakit dan gejala penyakit yang nampak pada daerah dada (Er, Yumusak, & Temurtas, 2010).

Seiring dengan kemajuan teknologi komputasi dan pengolahan *data*, *file* rekam medis seperti foto x-ray dapat disimpan menjadi *file* citra berekstensi umum seperti *'.jpeg'*, *'.jpg'*, *'.png'*, dan *'.bmp'* yang dapat dieksekusi/dibuka selayaknya *file* citra biasa pada *operation system* umum seperti Windows, MacOS, dan Linux.

CHEST X-RAY PASIEN COVID-19

Secara visual, hasil *CXR* dan *CT* pasien yang terinfeksi Covid-19 menunjukkan *anomaly* selayaknya pasien *pneumonia*, sebagian besar pasien menunjukkan *lung opacification*, *lesion* dan bercak-bercak *(paving)* (Xiang et al., 2020). Sebagai contoh, dapat dilihat pada Gambar 1.

Gambar 1 CXR pasien penderita Covid-19

Adapun karena ranah pengetahuan medis mengenai Covid-19 sendiri masih terbilang baru, belum terdapat definisi tetap mengenai gejala Covid-19 selain gejala mirip *flu*, kehilangan indra penciuman dan rasa, demam, dan *pneumonia*; selain *ground-glass opacification* (gejala ini akan nampak di kemudian hari setelah pasien masuk ke rumah sakit, bahkan pada pasien yang memiliki *CXR* awal masuk rumah sakit yang bersih atau tanpa gejala) (Cleverley, Piper, & Jones, 2020). Bahkan, menurut Cleverley et al. (Cleverley et al., 2020) sendiri, salah satu gejala Covid-19, yaitu *pneumonia* tidak begitu saja muncul pada kebanyakan pasien yang terjangkit Covid-19. Tetapi umumnya, sebagaimana kasus *pneumonia* lainnya, seiring waktu, akan terjadi peningkatan *lung density*, (atau pemutihan pada *CXR*) yang mungkin pada awalnya tidak terlihat karena tertunda.

GRAYSCALING

Sebagai *preprocessing*, citra/foto *CXR digital* berwarna atau *RGB triple-layered*, dirubah menjadi *grayscale image* sebelum *GLCM* diperoleh karena citra *input GLCM* haruslah berupa *grayscale image* 1 *layer* (2 dimensi).

Lebih lengkapnya, untuk merubah citra *triple layer RGB* menjadi *single layer grayscale*, dapat dilakukan dengan pelbagai cara (Kanan & Cottrell, 2012; Saravanan, 2010), yang umumnya adalah

memberikan bobot yang merupakan hasil aproksimasi selayaknya mata manusia mempersepsikan kecerahan.

Metode yang digunakan pada piranti pengolah grafis atau komputasi, seperti MATLAB, untuk memperoleh citra *grayscale* pada umumnya menggunakan metode yang disebut *'Luminance'*, dan diimplementasikan pada fungsi *built-in 'rgb2gray'*.

Luminance diperoleh dengan cara sebagai berikut,

$$G_{Luminance} = (0.3 * R) + (0.59 * G) + (0.11 * B) \quad (1)$$

Catatan,

$G_{Luminance}$ adalah citra *grayscale*, *R* adalah lapis pertama citra *RGB*, *G* adalah lapis kedua citra *RGB*, dan *B* adalah lapis ketiga citra *RGB*.

Alternatif dari metode ini yang sedikit lebih kompleks (Saravanan, 2010), adalah sebagai berikut,

$$Y = (0.299 * R) + (0.587 * G) + (0.114 * B) \quad (2)$$
$$U = (B - Y) * 0.565 \quad (3)$$
$$V = (R - Y) * 0.713 \quad (4)$$
$$UV = U + V \quad (5)$$
$$R1 = R * 0.299 \quad (6)$$
$$R2 = R * 0.587 \quad (7)$$
$$R3 = R * 0.114 \quad (8)$$
$$G1 = G * 0.299 \quad (9)$$
$$G2 = G * 0.587 \quad (10)$$
$$G3 = G * 0.114 \quad (11)$$
$$B1 = B * 0.299 \quad (12)$$
$$B2 = B * 0.587 \quad (13)$$
$$B3 = B * 0.114 \quad (14)$$
$$R4 = \frac{R1+R2+R3}{3} \quad (15)$$
$$G4 = \frac{G1+G2+G3}{3} \quad (16)$$
$$B4 = \frac{B1+B2+B3}{3} \quad (17)$$
$$I = (R4 + G4 + B4 + UV) \quad (18)$$

Pada (2), *R* adalah lapis pertama citra *RGB*, *G* adalah lapis kedua citra *RGB*, dan *B* adalah lapis ketiga citra *RGB*. *Y* adalah matriks *single layered* yang diambil dari citra original *RGB triple-layered*. *U* dan *V* pada (3) dan (4) adalah matriks *chrominance* dan *luminance* yang nanti akan dijumlah di (5) menjadi matirks *UV*.

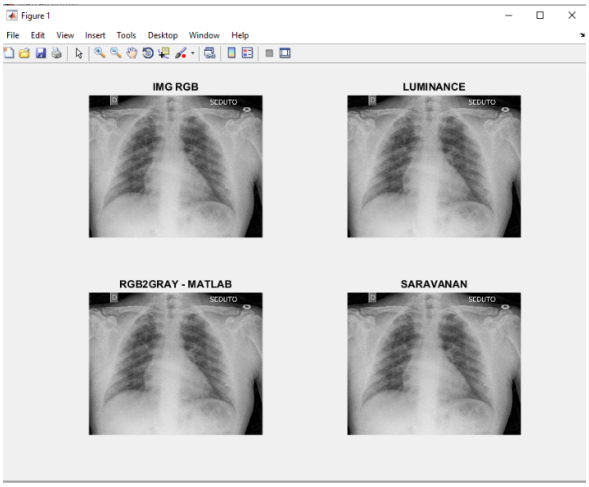
Persamaan (6) hingga (17) mengaproksimasi *RGB value* dari masing-masing *layer* citra asli menjadi matriks *R4*, *G4*, dan *B4*.

Barulah *I*, atau matriks *grayscaled image* dua dimensi dapat diperoleh dengan menjumlah matriks R4, G4, B4, dan UV pada persamaan (18).

Hasil dari metode *Luminance*, fungsi MATLAB *'rgb2gray'* dan metode Saravanan dapat dilihat pada Gambar 2.

GRAY-LEVEL CO-OCCURRENCE MATRICES

Gray-Level Co-Occurrence Matrices (GLCM) atau *Gray-Tone Spatial Dependence Matrices* adalah sebuah konsep yang diperkenalkan oleh R. M. (Id, Asklund, & Nyholm, 2019). Tujuannya adalah agar citra/foto dapat dianalisa, diperoleh suatu himpunan *textural feature* yang informatif, dikenali sebagai sebuah pola, dan diklasifikasikan.



Gambar 2. Perbandingan *CXR* pasien Covid-19 berekstensi *file '.jpeg'* dengan metode *Luminance*, *'rgb2gray'*, dan metode Saravanan

Ide dasar *GLCM* adalah bagaimana sebuah citra/foto hitam-putih *(grayscale image)* dapat dinyatakan sebagai fungsi dalam matriks persegi dua dimensi. *GLCM* diperoleh dengan menghitung banyaknya pasangan dua elemen *(pixel)* matriks (citra/foto hitam-putih) dengan arah tertentu (biasanya antara $0°, 45°, 90°, 135°$) yang *range* awalnya berkisar antara 0 hingga 255, dikuantisasi menjadi *positive integer* (1 hingga N).

Haralick et al. menotasikan perolehan matriks *GLCM*, dimisalkan matriks *I*, untuk satu himpunan $G = \{1,2,3, \dots, N_g\}$ yaitu nilai *grey-tone* yang telah dikuantisasikan, dengan N *positive integer* dengan $L_x = \{1,2,3, \dots, N_x\}$ sebagai *horizontal spatial domain* dan $L_y = \{1,2,3, \dots, N_y\}$ sebagai *vertical spatial domain*, maka himpunan I dapat dinyatakan sebagai pemetaan nilai G ke setiap pasang koordinat di $L_x \times L_y$, atau dengan kata lain $I: L_x \times L_y \rightarrow G$.

Fungsi pemetaan untuk arah 0° dan jarak antar pasangan elemen matriks sebesar *d*, didapat sebagai berikut,

$$p(i,j,d,0°) = \#\{((k,l),(m,n)) \in (L_x \times L_y) \times (L_y \times L_x) \mid k - m = 0,\ |l - n| = d,\ I(k,l) = i,\ I(m,n) = j\} \quad (19)$$

Dalam artikel aslinya, Haralick et al menuliskan 14 *textural features*, atau umumnya disebut dengan *Haralick Features* (Id et al., 2019; Loewke, 2013). Akan tetapi, pada umumnya, hanya 4 (Mahmood & Abbas, 2016) atau 5 (Brynolfsson et al., 2017; Zayed & Elnemr, 2015) dari 14 *feature* saja yang digunakan sebagai *feature descriptor*, yaitu *Contrast*, *Correlation*, *Energy (Angular Second Movement)*, *Homogenity*, dan *Entropy*. Piranti lunak komputasi MATLAB hanya mengkalkulasi 4 *feature* (Loewke, 2013) lewat fungsi *built-in* 'graycoprops'.

Berikutnya, kelima *Haralick Features* diperoleh sebagai berikut,

$$Contrast = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g}|i-j|^2 p(i,j) \quad (20)$$

$$Correlation = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g}\frac{p(i,j)ij-\mu_x\mu_y}{\sigma_x\sigma_y} \quad (21)$$

$$Energy\ (ASM) = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g}[p(i,j)]^2 \quad (22)$$

$$Homogenity = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g}\frac{p(i,j)}{1+(i-j)^2} \quad (23)$$

$$Entropy = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g}(-\ln p(i,j))p(i,j) \quad (24)$$

Catatan, *p(i,j)* adalah elemen matriks *GLCM*, dan nilai μ_x, μ_y, σ_x, dan σ_y didapat sebagai berikut,

$$\mu_x = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g} i\ p(i,j) \quad (25)$$

$$\mu_y = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g} j\ p(i,j) \quad (26)$$

$$\sigma_x = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g}\sqrt[2]{p(i,j)(j-\mu_x)^2} \quad (27)$$

$$\sigma_y = \sum_{i=1}^{N_g}\sum_{i=1}^{N_g}\sqrt[2]{p(i,j)\left(j-\mu_y\right)^2} \quad (28)$$

Kontras adalah ukuran ketergantungan *linear* dari *gray-level* untuk setiap pasang elemen matriks citra (*pixel*). Nilainya berada di antara 0 dan MIN(size(*GLCM*)).

Korelasi adalah ukuran saling keterkaitan setiap *pixel* dengan tetangganya. Nilainya berada di *interval* [-1, 1].

Energi adalah ukuran kereseragaman tekstur yang mengacu pada repetisi pasangan *pixel*. Nilainya berada di *interval* [0, 1].

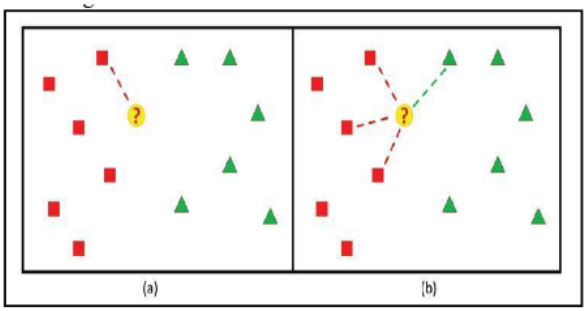
Homogenitas adalah ukuran keseragaman entri positif dari *GLCM*. Nilainya berada di *interval* [0, 1].

Entropi adalah kebalikan dari Energi.

HISTOGRAM OF ORIENTED GRADIENT

Konsep *Histogram of Oriented Gradient* (HOG) adalah konsep *descriptor* yang dipopulerkan oleh Dalal and Triggs (Dalal & Triggs, 2005) untuk *human detection*. Setelah itu, HOG digunakan untuk identifikasi objek secara umum seperti daun tanaman, kendaraan, tulisan tangan, hingga objek medis seperti tumor.

HOG diperoleh dengan menghitung *magnitude* dan *gradient* untuk setiap elemen/*pixel* pada citra yang dipartisi menjadi blok sejumlah $n \times m$ dan rentang *gradient* tertentu, umumnya antara 0° hingga 180° dengan interval 20° (180/20 = 9 *bin*) atau 45° (4 *bin*).



Gambar 3. (a) kerja decision rule dari 1-NN, (b) kerja decision rule dari 4-NN

Umumnya, *gradient* pada citra diperoleh dengan konvolusi menggunakan matriks *filter*. Salah satu *filter* yang umum digunakan adalah *filter* Sobel.

Sehingga untuk citra *A*, dinyatakan sebagai matriks *A(x,y)* (dan simbol $*$ untuk konvolusi),

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (29)$$

Maka, dengan menggunakan matriks Sobel,

$$\frac{\partial f}{\partial x} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} * A \quad (30)$$

$$\frac{\partial f}{\partial y} = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix} * A \quad (31)$$

Lalu, arah (sudut θ) dan k *(magnitude)* dapat dihitung dengan,

$$\theta = tan^{-1}\left[\frac{g_y}{g_x}\right] \quad (32)$$

$$k = \sqrt[2]{g_y^2 + g_x^2} \quad (33)$$

K-NEAREST NEIGHBOR

K–Nearest Neighbor (k-NN), adalah metode klasifikasi *non parametric* (disebut juga *case based reasoning*) yang bekerja dengan cara 'menemukan' *output* dari kasus lampau atau 'memori' yang ada (Alpaydin, 2010). Sederhananya, sebuah data diklasifikasikan menurut kelas k-tetangga terdekat data tersebut. *K-Nearest Neighbor* adalah metode yang paling sederhana yang digunakan bila pengetahuan *priori* mengenai distribusi data adalah sedikit diketahui atau tidak dikatahui sama sekali, sehingga sampel yang tidak diketahui labelnya dapat dikllasifikasikan (Imandoust & Bolandraftar, 2013).

Karena sifat k-NN yang *simple*, dan *transparent*, k-NN adalah metode klasifikasi *non-parametric* yang

paling umum digunakan (Alpaydin, 2010; Cunningham & Delany, 2007; Imandoust & Bolandraftar, 2013). Performa *k-NN* bergantung pada pemilihan nilai k, dan *metric* yang digunakan (Imandoust & Bolandraftar, 2013), dan metric yang memenuhi sifat *metric*; *non-negativity, identity, symmetry, triangle inequality*, yang umum dipakai adalah metric Euclidean, Mahalanobian, Cosine, dan Cityblock.

Ilustrasi pada Gambar 3 menjelaskan bagaimana cara kerja *k-NN* pada *metric* Euclid.

Aplikasi dari k-NN terdapat di pelbagai bidang keilmuan, seperti baahasa, agrikultur, keuangan, dan medis (Imandoust & Bolandraftar, 2013).

METODE

Pada penelitian ini , metode yang digunakan dalam penelitian ini adalah sebagai berikut :

DATASET

Dataset citra *CXR digital* yang digunakan adalah *subset* dari *open-access dataset* COVID-XRay-5K Dataset (*link* menuju situs *master* , https://github.com/shervinmin/DeepCovid/tree/master/data) yang berisikan 5000 *CXR* posisi *Anterior-Posterior* yang digunakan dalam penelitian artikel *Deep-COVID : Predicting COVID-19 From Chest X-Ray Images Using Deep Transfer Learning* (Minaee, Kafieh, Sonka, Yazdani, & Jamalipour, 2020).

Situs GitHub memberikan *dataset* berupa *compressed file* berkestensi *'.zip'* dengan nama *'data_upload_v3.zip'* yang dapat didekompresi setelah diunduh. Ekstensi dari *file* citra *CXR* pada *folder dataset* (baik *folder train* maupun *folder test*) adalah bervariasi antara *format '.jpeg'*, *'.jpg'* dan *'.png'*.

PREPROCESSING

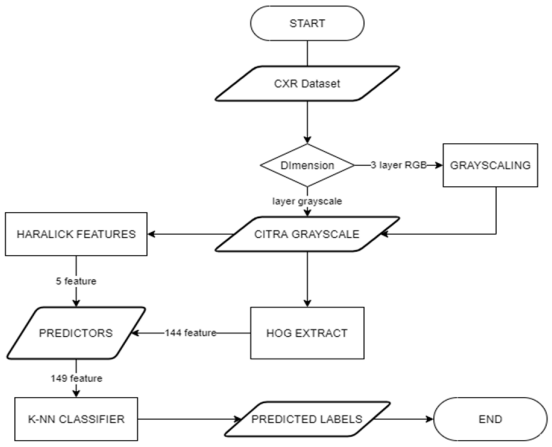
Dataset yang diambil dari 5000 citra adalah 1000 citra yang sudah melalui *preprocessing grayscaling* dengan label "0" atau kelas *negative* sejumlah 820 (420 *No Finding*, 200 *Lung Opacity*, 200 *Pneumonia*) dan sisanya berlabel "1" atau kelas *positive*. Partisi *dataset* menjadi *training set* dan *test set* dilakukan secara *random sampling* dengan rasio 4:1 demi mencegah *overfitting* (Gholamy, Kreinovich, & Kosheleva, 2018). Citra yang ditemui *corrupt* pada *folder* di *system file* dihapus.

Jumlah *feature* yang digunakan sebagai *predictor* berjumlah 149 *feature* yang terdiri dari 5 *texture feature GLCM*, yaitu *Contrast*, *Correlation*, *Energy (Angular Second Movement)*, *Homogenity*, dan *Entropy*, dari 14 *feature* yang ada (Zayed & Elnemr, 2015) sebagai hasil *feature selection* dan 144 *HOG feature* dari pembagian CXR menjadi blok sejumlah 4×4 dan rentang *gradient* dari 0° hingga 180° dengan *interval* 20° (180/20 = 9 *bin*).

KLASIFIKASI K-NEAREST NEIGHBOR

Nilai k dapat diestimasi dengan cara mengakar kuadratkan n-jumlah anggota *training data*, atau setengah dari nilai tersebut, hingga nilai k terbaik ditemukan (Nadkarni, 2016).

Dari estimasi awal $k = \sqrt[2]{800} \approx 28$, dibagi dua, dan di coba hingga diperoleh nilai k terbaik dan hasil *k-NN*-nya pada ke-empat *metric*.



Gambar 4. Diagram Alur Metode Penelitian

Hasil eksperimen akan dievaluasi menggunakan bantuan *confusion matrix* untuk menghitung *accuracy* dari *classifier*.

Eksperimen dilakukan pada komputer dengan *processor* Core-i5 4200U, *Graphic Processing Unit* Nvidia GT720M 2GB dan memori *RAM* 8GB. Piranti lunak yang digunakan adalah MATLAB 2016b.

PROSES EVALUASI HASIL KLASIFIKASI

Evaluasi dilakukan dengan bantuan *confusion matrix*, yaitu matriks persegi yang elemen-elemennya mendeskripsikan perfoma dari suatu *classifier* dengan membandingkan jumlah label *data* hasil *output classifier* dan label *data* aslinya.

Tabel 1. *Confusion Matrix*

	PREDICTED	
TRUE	TP	FN
	FP	TN

Hasil klasifikasi akan terbagi menjadi TP (*True Positive*), TN (*True Negative*), FP (*False Positive*), dan FN (*False Negative*), yang dapat dilihat dari *confusion matrix* di Tabel 1.

True Positive berarti label data *output classifier* sesuai label aslinya, yaitu *positive* atau '1'. *True Negative* berarti, label data *output classifier* sesuai label

aslinya, yaitu *negative* atau '0'. *False Positive* berarti data yang memiliki label asli '0' atau *negative*, terklasifikasikan sebagai *positive*, dan *False Negative* adalah sebaliknya, yaitu data yang memiliki label asli '1' atau *positive*, terklasifikasikan sebagai *negative.*

Terhadap jumlah seluruh sampel (N), relasi antara N dengan *TP*, *TN*, *FP*, dan *FN*, adalah sebagai berikut,

$$N = TP + TN + FP + FN \quad (34)$$

Adapun cara untuk memperoleh hasil seperti pada Tabel 1 adalah sebagai berikut,

$$Accuracy = \frac{TP+TN}{N} \quad (35)$$

yaitu banyaknya data yang teridentifikasi benar dibandingkan jumlah seluruh sampel.

Pada implementasinya, *confusion matrix* yang terdapat pada program komputasi MATLAB sebagai fungsi *built-in 'confmat'*.

PEMBAHASAN

Pada tahap ini, telah dilakukan ujicoba pada *dataset* 1000 *CXR* yang dilewatkan *preprocessing* berupa proses *grayscaling* (1-18), perolehan *Haralick Features* (19-28) dan *HOG descriptor* (29-33) sehingga *dataset* dapat direpresentasikan sebagai matriks *feature vectors* berordo 1000*149, yang akan dipartisi menjadi 80% untuk *training set* dan sisanya untuk *test set* menggunakan *random sampling*, dengan nilai k yang berbeda untuk *metric* Euclid, Mahalanobian, Cosine, dan Cityblock.

Hasil dari proses klasifikasi dievaluasi menggunakan *confusion matrix* seperti pada Tabel 1 dan ukuran berupa akurasi (34-35) dengan bantuan piranti lunak MATLAB.

Tabel 2. Hasil Klasifikasi *k-NN*

k	Accuracy			
	Euclid	**Maha.**	**Cosine**	**Cityblock**
13	99%	92%	99%	99.5%
12	99%	93%	98.5%	99.5%
11	98.5%	94%	98.5%	99.5%
10	99.5%	93%	99%	99.5%
9	98.5%	93%	98.5%	99%
8	98.5%	92.5%	98.5%	99.5%

Tabel 2 menunjukkan performa *classifier k-NN* dalam mengklasifikasikan *CXR* pasien yang menderita Covid-19 dan yang tidak.

Diperoleh bahwa hasil terbaik adalah rata-rata akurasi dengan k=10 (k=11 bila k yang diinginkan ganjil demi menghindari kasus imbangnya jumlah kedua label *neighbor*, yaitu 5 label '0' dan 5 label '1') yang terdapat pada Tabel 2 untuk 4 jenis *metric*.

Metric Mahalanobis memiliki performa terendah diantara ke-empat *metric* yang digunakan. Ini karena *metric* Mahalanobis menggunakan *covariance matrix*, yang berarti 'jarak' dua sampel yang memiliki korelasi tinggi akan jadi 'lebih pendek', yang secara konsekuen, *Mahalanobian distance* lebih sensitif terhadap *outlier* pada distribusi sampel. Singkatnya, *metric* Mahalanobian adalah cocok untuk menemukan apakah ada *outlier* pada *sample space*. Komputasi untuk *metric* Mahalanobis adalah yang paling sukar untuk dilakukan dibanding *metric* Euclid, Cosine, dan Cityblock.

Hasil *confusion matrix* dan akurasi pada Tabel 2 dapat bervariasi, bergantung dari *random sampling*. Akan tetapi eksekusi program berulangkali menunjukkan hasil yang tidak jauh berbeda, yaitu di atas rata-rata 90%.

Pada *dataset* asli, *cut-off* sebanyak 4000 citra *CXR* dilakukan sebagai usaha untuk mencegah ketimpangan data, karena jumlah citra yang dilabeli positif "Covid-19" hanya berjumlah 184, dimana 4 file citra yang *corrupt* dihapus demi menghilangkan *outlier* sehingga didapat hanya 180 citra *CXR* saja yang berlabel "Covid-19". Kemudian, *CXR* dengan gejala dipilih selain "*No Finding* / Sehat tanpa gejala" adalah "*Pneumonia*" dan "*Lung Opacity*" karena memiliki ciri gejala yang mirip secara empiris dengan *CXR* pasien Covid-19 (Cleverley et al., 2020; Xiang et al., 2020).

Sebagai diskusi tambahan, diketahui secara intuitif, *k-NN* memiliki interpretabilitas yang transparan, algoritma yang sederhana, sehingga baik implementasi dan proses *debugging* mudah dilakukan. Sedangkan sebagai kelemahan, *k-NN* sangat sensitif terhadap *outlier*, dan pada *redundant features* sehingga *feature selection* seperti pengambilan *feature descriptor* yang sama dengan penelitian terdahulu (Zayed & Elnemr, 2015) atau penggunaan *Principal Component Analysis* dapat dilakukan karena semua *feature* berkontribusi pada kalkulasi 'jarak'. Terlebih lagi, *k-NN* bisa saja memiliki *run-time* yang lama karena merupakan *instance-based learning*, terlebih untuk *dataset* yang berukuran besar (Cunningham & Delany, 2007).

PENUTUP

SIMPULAN

Berdasarkan hasil pembahasan disimpulkan bahwa penerapan dari *Machine Learning* dapat digunakan menyelesaikan permasalahan riil, yaitu untuk melakukan *image classification*. Klasifikasi *k-Nearest Neighbour* menggunakan *Haralick Features*

dan *Histogram of Oriented Gradient* dengan jumlah 149 *features* pada 1000 *CXR* pada *dataset* COVID-Xray-5k dengan 180 (dari 184 citra *CXR*, dengan 4 citra dihapus karena *corrupt*) diantaranya adalah *CXR* pasien yang terinfeksi Covid-19, menghasilkan akurasi rata-rata dari Tabel 1 di atas 90%, dengan hasil terbaik diperoleh dengan nilai k=10.

Performa terendah adalah pada *metric* Mahanalobian, yaitu 93%, dan akurasi terbaik pada *metric* Euclid dan Cosine, yaitu 99.5%. Akurasi yang didapat dipengaruhi oleh keberadaan *outlier* pada *dataset*.

SARAN

Saran bagi penelitian kedepannya adalah mencoba membandingkan hasil kinerja classifier *Machine Learning k-Nearest Neighbour* dengan penggunaan *feature descriptor* yang berbeda selain 'tekstur' dapat digunakan. Karena minimnya jumlah *CXR* pasien Covid-19 yang tersedia untuk umum, penelitian lanjutan harus dilakukan pada *dataset* yang memiliki jumlah *CXR* pasien Covid-19 lebih banyak demi menguji keakuratan *classifier*.

DAFTAR PUSTAKA

Alpaydin, E. (2010). *Introduction to Machine Learning* (2nd ed.). Cambridge: MIT Press.

Barry, O. De, Obadia, I., El Hajjam, M., & Carlier, R. (2020). Chest-X-ray is a mainstay for follow-up in critically ill patients with covid-19 induced pneumonia. *European Journal of Radiology*, *129*(May).https://doi.org/10.1016/j.ejrad.2020.109075

Brynolfsson, P., Nilsson, D., Torheim, T., Ask, T., Thellenberg, C., Trygg, J., … Garpebring, A. (2017). Haralick texture features from apparent diffusion coefficient (ADC) MRI images depend on imaging and pre-processing parameters. *Scientific Reports*, *7*(4041), 1–11. https://doi.org/10.1038/s41598-017-04151-4

Cleverley, J., Piper, J., & Jones, M. M. (2020). The role of chest radiography in confirming covid-19 pneumonia. *British Medical Journal - Practice Pointer*, *370* (2426). https://doi.org/10.1136/bmj.m2426

Cunningham, P., & Delany, S. J. (2007). *k -Nearest Neighbour Classifiers*. Dublin, Ireland.

Daffner, R. H., & Hartman, M. S. (2013). *Clinical Radiology : The Essentials* (4th ed.). Philadelphia: Lippincolt William & Wilkins.

Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection.

Er, O., Yumusak, N., & Temurtas, F. (2010). Chest diseases diagnosis using artificial neural networks. *Expert System Application*, *37*(12), 7648–7655. https://doi.org/10.1016/j.eswa.2010.04.078

Farhan, A. H., & Kamil, M. Y. (2020). Texture Analysis of Mammogram Using Histogram of Oriented Gradients Method Texture Analysis of Mammogram Using Histogram of Oriented Gradients Method. https://doi.org/10.1088/1757-899X/881/1/012149

Gholamy, A., Kreinovich, V., & Kosheleva, O. (2018). *Why 70 / 30 or 80 / 20 Relation Between Training and Testing Sets : A Pedagogical Explanation*. El Paso, Texas.

Id, P. B., Asklund, T., & Nyholm, T. (2019). Gray-level invariant Haralick texture features. *PLOS ONE*, 1–18. https://doi.org/https://doi.org/10.1371/journal.pone.0212110

Imandoust, S. B., & Bolandraftar, M. (2013). Application of K-Nearest Neighbor (KNN) Approach for Predicting Economic Events : Theoretical Background. *International Journal of Engineering Research and Applications*, *3*(5), 605–610. Retrieved from https://www.researchgate.net/publication/304826093_Application_of_K-nearest_neighbor_KNN_approach_for_predicting_economic_events_theoretical_background

Iwahori, Y., Hattori, A., Adachi, Y., Bhuyan, M. K., & Robert, J. (2015). Automatic Detection of Polyp Using Hessian Filter and HOG Features. *Procedia - Procedia Computer Science*, *60*, 730–739. https://doi.org/10.1016/j.procs.2015.08.226

Jacobi, A., Chung, M., Bernheim, A., & Eber, C. (2020). Portable chest X-ray in coronavirus disease-19 (COVID-19): A pictorial review. *Clinical Imaging*, *64*(January), 35–42.

Jaeger, S., Karargyris, A., Jaeger, S., Karargyris, A., Candemir, S., Folio, L., … Mcdonald, C. J. (2014). Automatic Tuberculosis Screening Using Chest Radiographs. *IEEE Transactions on Medical Imaging*, *33*(2), 233–245. https://doi.org/10.1109/TMI.2013.2284099

Kanan, C., & Cottrell, G. W. (2012). Color-to-Grayscale : Does the Method Matter in Image Recognition ? *PLOS ONE*, *7*(1). https://doi.org/10.1371/journal.pone.0029740

Liu, Y., Kuo, R., & Shih, S. (2020). COVID-19 : The first documented coronavirus pandemic in history. *Biomedical Journal*, (xxxx), 1–6. https://doi.org/10.1016/j.bj.2020.04.007

Loewke, N. (2013). *Haralick Texture Analysis for Stem Cell Identification*. Stanford. Retrieved from https://stacks.stanford.edu/file/druid:np318ty6250/Loewke_Stem_Cell_Identification.pdf

Mahmood, F. H., & Abbas, W. A. (2016). Texture Features Analysis using Gray Level Co-occurrence Matrix for Abnormality Detection in Chest CT Images Texture Features Analysis using Gray Level Co-occurrence Matrix for Abnormality Detection in Chest CT Images ةيعطقملا روص لا يف ذوذشلا فشكل (*GLCM*. *Iraqi Journal of Science*, *57*(1A), 279–288.

Minaee, S., Kafieh, R., Sonka, M., Yazdani, S., & Jamalipour, G. (2020). Deep-COVID: Predicting COVID-19 From Chest X-Ray Images Using Deep Transfer Learning. *Medical Image Analysis*, 1–9. https://doi.org/https://doi.org/10.1016/j.media.2020.101794

Nadkarni. (2016). CHAPTER 10 - Core Technologies : Data Mining and " Big Data ." In *Clinical Research Computing* (pp. 187–204). https://doi.org/10.1016/B978-0-12-803130-8/00010-5

Ruttens, D., Kerrebroeck, M. Van, Vandewalle, J., Falter, M., Daenen, M., Thevissen, K., … Thomeer, M. (2020). Respiratory Medicine Case Reports Fever and an abnormal chest X-ray during the COVID-19 pandemic. *Respiratory Medicine Case Reports*, *31*, 101167. https://doi.org/10.1016/j.rmcr.2020.101167

S, S. K. P., & Dharun, V. S. (2016). Extraction of Texture Features using GLCM and Shape Features using Connected Regions. *International Journal of Engineering and Technology*, *8*(6), 6–11. https://doi.org/10.21817/ijet/2016/v8i6/160806254

Saravanan, C. (2010). Color Image to Grayscale Image Conversion Color Image to Grayscale Image Conversion. In *Second Conference on Computer Engineering and Applications*. https://doi.org/10.1109/ICCEA.2010.192

Saunders, B. F. (2008). *CT Suite : The Work of Diagnosis in the Noninvasive Cutting*. Durham, North Carolina: Duke University Press.

World Health Organization Emergency Dashboard. (2021). WHO Coronavirus Disease (COVID-19) Dashboard. Retrieved January 13, 2021, from https://covid19.who.int/

Xiang, C., Lu, J., Zhou, J., Guan, L., Yang, C., & Chai, C. (2020). Research Article CT Findings in a Novel Coronavirus Disease (COVID-19) Pneumonia at Initial Presentation. *BioMedical Research International*. https://doi.org/https://doi.org/10.1155/2020/5436025

Zayed, N., & Elnemr, H. A. (2015). Statistical Analysis of Haralick Texture Features to Discriminate Lung Abnormalities. *International Journal of Biomedical Imaging*. https://doi.org/http://dx.doi.org/10.1155/2015/267807

Zulpe, N., & Pawar, V. (2012). GLCM Textural Features for Brain Tumor Classification. *International Journal of Computer Science Issues*, *9*(3), 354–359.