

Identifikasi Batu Ginjal Menggunakan Metode Convolutional Neural Network

Ilham Yori Pradana^{*1)}, Ardian Yusuf W. S.Kom., M.Kom.²⁾, M. Sholik. S.Kom., M.Kom.³⁾

¹⁾Rekayasa Perangkat Lunak, Fakultas Teknologi Infomasi & Bisnis, Institut Teknologi Telkom Surabaya, Jl.
Ketintang No. 156, Ketintang, Kec. Gayungan, Surabaya, 60231, Negara
ilhamyp@student.ittelkom-sby.ac.id

Abstrak

Arus globalisasi membawa dampak yang besar salah satu dampaknya adalah meningkatnya kasus-kasus penyakit, dan salah satunya adalah batu ginjal. Batu ginjal terjadi karena suatu zat dalam darah yang mengkristal pada ginjal. Efek dari batu ginjal tersebut adalah infeksi, penyumbatan aliran kencing dan dapat merusak ginjal, menurut data prevalensi 6 dari 1000 penduduk Indonesia merupakan penderita batu ginjal. Maka untuk membantu dalam pendeteksian dini maka diterapkan metode Convolutional Neural Network untuk membantu dalam mendiagnosa dini penyakit tersebut. Arsitektur sistem dari model tersebut nantinya akan seperti berikut, dataset akan melalui proses pre-processing menggunakan CLAHE, kemudian proses training berjalan menggunakan CNN dan akan berjalan didalam stratified k-fold cross validation, k-fold digunakan untuk mengukur kualitas model. Hasil dari penelitian ini didapatkan bahwa model yang menggunakan Convolutional Neural Network, memiliki performa lebih baik dibandingkan pada penelitian sebelumnya, performa model ini memiliki akurasi 85% sedangkan pada penelitian sebelumnya yang menggunakan model Extreme Learning Machine mendapatkan 80,76%.

Kata kunci: *Batu Ginjal, Convolutional Neural Network, Ultrasonografi*

1. Pendahuluan (Introduction)

Meningkatnya arus globalisasi membawa dampak perubahan yang besar. Dampak positif tersebut dapat dilihat dari perubahan gaya hidup, perilaku, dan lingkungan, selain itu dampak negatif yang dihasilkan adalah meningkatnya kasus-kasus penyakit, dan salah satunya adalah batu ginjal. Ginjal sendiri merupakan organ ekskresi pada manusia yang berbentuk mirip dengan kacang yang berfungsi untuk menyaring kotoran yang masuk dari darah dan membuangnya bersama dengan urine. Batu ginjal terjadi karena mineral atau zat lain dalam darah yang mengkristal pada ginjal dan membentuk suatu padatan. Jika hal tersebut dibiarkan batu ginjal mampu menimbulkan masalah seperti infeksi, penyumbatan aliran kencing bahkan dapat merusak organ ginjal, menurut data prevalensi 6 dari 1000 penduduk Indonesia merupakan penderita batu ginjal (Universitas Airlangga News, 2020).

Ada beberapa metode yang dapat digunakan untuk mendeteksi penyakit batu ginjal seperti tes darah, foto rontgen, *CT-Scan* dan salah satunya *USG*. *USG* atau ultrasonografi merupakan metode pemindaian dengan memanfaatkan gelombang suara berfrekuensi tinggi untuk menghasilkan gambar dari kondisi organ atau jaringan dalam tubuh. Agar dapat mengidentifikasi penyakit batu ginjal biasanya dokter akan mengumpulkan informasi gejala yang dialami oleh pasien, kemudian akan dilanjutkan dengan metode yang telah disebutkan sebelumnya. Tetapi metode yang sering digunakan adalah *USG* karena harganya yang lebih terjangkau, tidak memerlukan persyaratan khusus, dan juga tidak menyebabkan efek samping.

Telah ada penelitian yang dilakukan untuk mengidentifikasi penyakit melalui citra ultrasonografi. Penelitian tersebut melalui beberapa tahap yaitu, image pre-processing yang menggunakan *Contrast Limited Adaptive Histogram Equalization (CLAHE)*, setelah itu dilakukan segmentation yang menggunakan metode *threshold* otsu dan *morphological close*, dan dilanjutkan dengan *feature*

extraction menggunakan *Gray Level Co-occurrence matrix*, dan akhirnya dilakukan identifikasi *Extreme Machine Learning* yang mendapatkan nilai akurasi 80,76% [9].

Mengingat latar belakang yang sudah dijabarkan, dan pada penelitian sebelumnya masih belum ada yang melakukan penelitian menerapkan *convolutional neural network* pada gambar ultrasonografi ginjal, oleh karena itu penelitian ini dilakukan untuk mengetahui seberapa efektif metode *convolutional neural network* dan juga diharapkan dapat mempermudah pendeteksian dini penyakit batu ginjal.

2. Metode Penelitian (Methods)

2.1. Pre-Processing

Image pre-processing adalah sebuah metode yang dilakukan pada gambar yang bertujuan untuk meningkatkan kualitas pada gambar hal tersebut dilakukan dengan cara menekan *noise* dan distorsi pada gambar atau meningkatkan fitur yang penting pada gambar tersebut (Sonka, et al., 2014).

2.1.1. Resize

Resizing adalah proses merubah ukuran suatu objek menjadi lebih besar atau lebih kecil, dalam *resizing* terdapat dua metode yaitu *scaling* dan *cropping*, *scaling* merupakan metode yang digunakan untuk memperbesar dan memperkecil citra dengan fungsi interpolasi, sedangkan *cropping* bekerja dengan cara mengambil suatu area dari sebuah citra (Irawan, 2010).

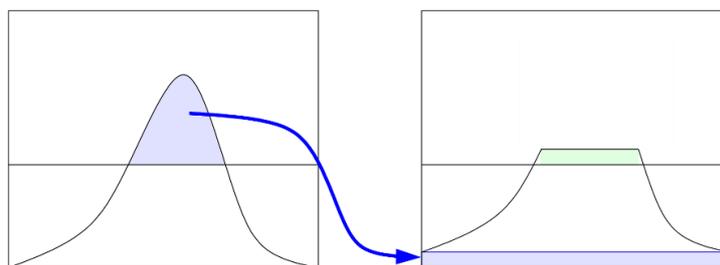
2.1.2. Contrast Limited Adaptive Histogram Equalization (CLAHE)

Metode *Contrast Limited Adaptive Histogram Equalization* merupakan metode yang dikembangkan dari metode *Adaptive Histogram Equalization*, kelebihan pada metode ini adanya *contrast limiting* atau *clip-limit* yang memudahkan kita mengatur batas maksimum *contrast* pada histogram (Yadav, et al., 2014) (Reza, 2004). Nilai batas tersebut dapat dihitung menggunakan persamaan berikut :

$$\beta = \frac{M}{N} \left(1 + \frac{\alpha}{100} (S_{max} - 1) \right)$$

- M : Luas region
- N : Nilai *grayscale*
- α : *Clip factor* batas limit histogram (1 - 100)
- S_{max} : Slope max

Histogram yang memiliki *pixel* dari *clip limit* nantinya akan dipotong dan hasil potongan tersebut akan didistribusikan kepada area disekitarnya yang berada dibawah *clip limit* (Reza, 2004). Proses tersebut dapat digambarkan seperti berikut ini :



Gambar 1 Penyebaran Region Histogram

Metode ini digunakan untuk meningkatkan *contrast* dari gambar ultrasonografi yang dimana sering memiliki bintang *noise* dan juga area yang memiliki *contrast* yang rendah (Singh, et al., 2020).

2.2. Augmentation

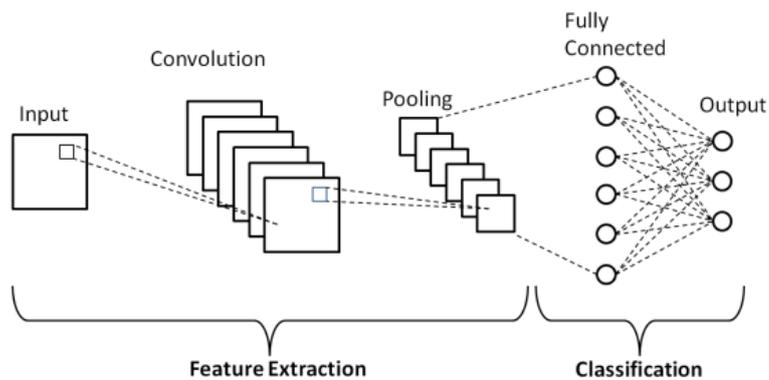
Data augmentation merupakan sebuah metode untuk membuat dataset lebih beragam dan seimbang antar kelasnya, karena ketika dataset mencukupi dari segi kuantitas dan kualitas hal tersebut dapat meningkatkan performa dan juga akurasi pada model *machine learning* [19].

Pada *data augmentation* terdapat dua pendekatan yaitu pendekatan dasar dan tingkat lanjut. Metode pendekatan tingkat dasar contohnya seperti *image manipulation*, *image erasing*, dan *image mix*, sedangkan metode tingkat lanjut yaitu *auto augment*, *feature augmentation*, dan *deep generative model* [19].

2.3. Convolutional Neural Network

Convolutional Neural Network merupakan metode yang dikembangkan dari *Multilayer Perceptron*, tetapi yang membedakan adalah neuron pada metode ini neuron disusun menjadi tiga dimensi yaitu *weight*, *length* dan *depth*, *depth* disini bukanlah jumlah layer tetapi volume aktivasi [20] [21].

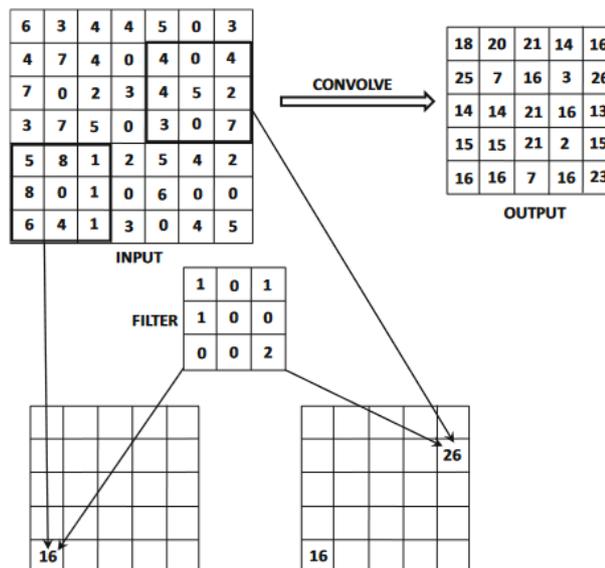
Desain tersebut CNN dituangkan kedalam algoritma seperti berikut. Pada saat CNN menerima input gambar, gambar tersebut akan melalui beberapa layer yang akhirnya akan diklasifikasikan kedalam satu kelas grup. Layer yang dilalui antara lain adalah *convolutional layer*, *pooling layer*, dan *fully connected layer* [20].



Gambar 2 Convolutional Neural Network

2.3.1. Convolutional Layer

Convolutional layer digunakan untuk melakukan operasi konvolusi dari layer sebelumnya, tujuan dari adanya *convolutional layer* ini adalah untuk mendeteksi *feature* seperti *edges*, *lines*, dan *color blobs*, dalam proses ini terdapat filter atau kernel yang digunakan mendeteksi *feature* tersebut, filter tersebut akan bergerak dari sudut kiri atas objek, menuju sudut kanan bawah objek, hasil dari proses tersebut dikenal sebagai *feature map* atau *convolved feature* [22] [23].

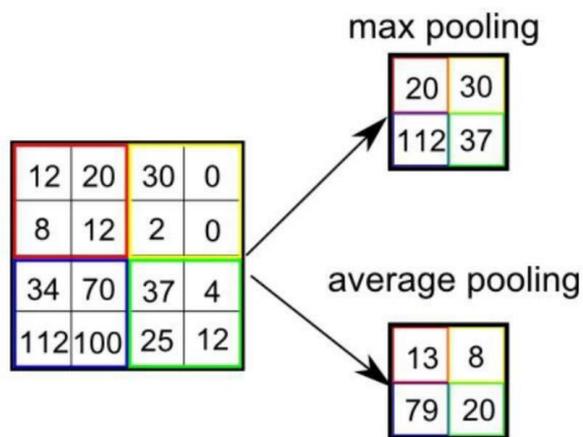


Gambar 3 Convolutional layer

Setelah mendapat *feature map*, akan dilanjutkan dengan menerapkan *Non-Linearity Activation Function*, salah satu fungsinya bernama *Rectified Linear Unit* atau ReLU, penerapan proses ini digunakan untuk mengenalkan non-linearitas pada jaringan yang fungsinya untuk membuat model dapat mempelajari pola yang kompleks dan membuat prediksi yang akurat [22] [23].

2.3.2. Pooling Layer

Pooling layer berfungsi sebagai *down sampling* dari *feature map* yang telah ada, tujuan dari *pooling layer* adalah untuk mengurangi dimensionalitas matriks, akibat dari hal tersebut adalah jumlah parameter berkurang, model tidak terlalu kompleks, dan mengurangi daya komputasi yang digunakan, terdapat dua jenis *pooling*, *Max pooling* yang mengembalikan nilai maksimum dari bagian objek yang dicakup kernel, dan *average pooling*, yang mengembalikan rata-rata semua nilai dari objek yang dicakup kernel. [20] [23].



Gambar 4 Pooling Layer

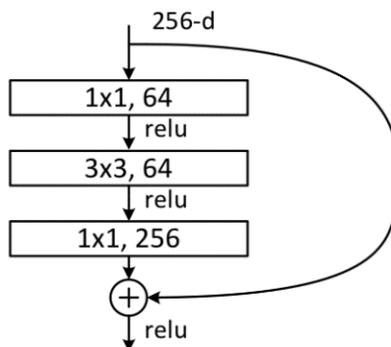
2.3.3. Fully Connected Layer

Pada *layer* ini setiap neuron saling terhubung ke tiap *layer* sebelumnya dan tidak akan terhubung ke *layer* milik mereka sendiri. *Layer* ini sering digunakan pada tahap akhir dari *neural network*, fungsi ini bertanggung jawab untuk memproses *feature*, dan membuat prediksi berdasarkan *feature* tersebut, sehingga dapat dikatakan bahwa lapisan ini dapat menangkap hubungan kompleks antar *feature* [23].

2.4. Residual Network 50

Residual Network atau ResNet merupakan *convolutional neural network*, yang diciptakan untuk mengatasi masalah mengenai *vanishing & exploding gradient*, kata 50 sendiri pada ResNet merupakan jumlah *layer* atau kedalaman dari jaringan tersebut. [23] [24].

Vanishing & exploding gradient merupakan masalah yang terjadi ketika ingin memperdalam suatu jaringan, maka dari itu ResNet memperkenalkan konsep *shortcut connections* atau *skip connections*, ia bekerja dengan cara menggunakan input dan output pada *layer* sebelumnya, digunakan sebagai input pada *layer* selanjutnya untuk menghasilkan output baru [25]. Berikut ini adalah gambaran dari *skip connections*.



Gambar 5 Skip Connections

Struktur dari ResNet50 akan seperti berikut, ia memiliki *convolutional layer* beserta *skip connections, pooling layer, dan full connected layer*. Setelah proses tersebut selesai pada bagian akhir ia akan menggunakan fungsi *softmax* [24].

| layer name | output size | 18-layer | 34-layer | 50-layer | 101-layer | 152-layer |
|------------|-------------|---|---|---|--|--|
| conv1 | 112×112 | 7×7, 64, stride 2 | | | | |
| | | 3×3 max pool, stride 2 | | | | |
| conv2_x | 56×56 | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ |
| conv3_x | 28×28 | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$ |
| conv4_x | 14×14 | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$ |
| conv5_x | 7×7 | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ |
| | 1×1 | average pool, 1000-d fc, softmax | | | | |
| FLOPs | | 1.8×10 ⁹ | 3.6×10 ⁹ | 3.8×10 ⁹ | 7.6×10 ⁹ | 11.3×10 ⁹ |

Gambar 6 Struktur *Residual Network*

2.5. Performance Evaluation Measure

Perfromance evaluation measure adalah satu tahapan untuk menguji atau mengukur performa dari suatu sistem atau model, salah satu metode yang dapat digunakan untuk mengukur performa suatu model adalah *confusion matrix* [26].

2.5.1. Confusion Matrix

Confusion matrix merupakan alat yang digunakan untuk melakukan analisis pada classifier apakah ia dapat mengenali label suatu kelas [27]. *confusion matrix* dapat digambarkan seperti berikut.

| | | Predicted class | | |
|--------------|-----|-----------------|----|-------|
| | | yes | no | Total |
| Actual class | yes | TP | FN | P |
| | no | FP | TN | N |
| Total | | P' | N' | P + N |

Gambar 7 *Confusion Matrix*

- *True Positive (TP)* : nilai prediksi positif dengan kondisi aktual positif
- *False Positive (FP)* : nilai prediksi positif dengan kondisi aktual negatif
- *False Negative (FN)* : nilai prediksi negatif dengan kondisi aktual negatif
- *False Positive (FP)* : nilai prediksi negatif dengan kondisi aktual positif

Gambar diatas merupakan tabel yang dapat digunakan untuk menganalisa seberapa bagus model yang digunakan, dari tabel diatas kita dapat menganalisa *accuracy, precision, recall, dan F Measure* [27]. Berikut ini adalah persamaan yang digunakan untuk mengukur 4 hal tersebut.

$$Accuracy = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + FP + FN + TN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN} = \frac{TP}{P}$$

$$F\ Score = \frac{2 \times precision \times recall}{precision + recall}$$

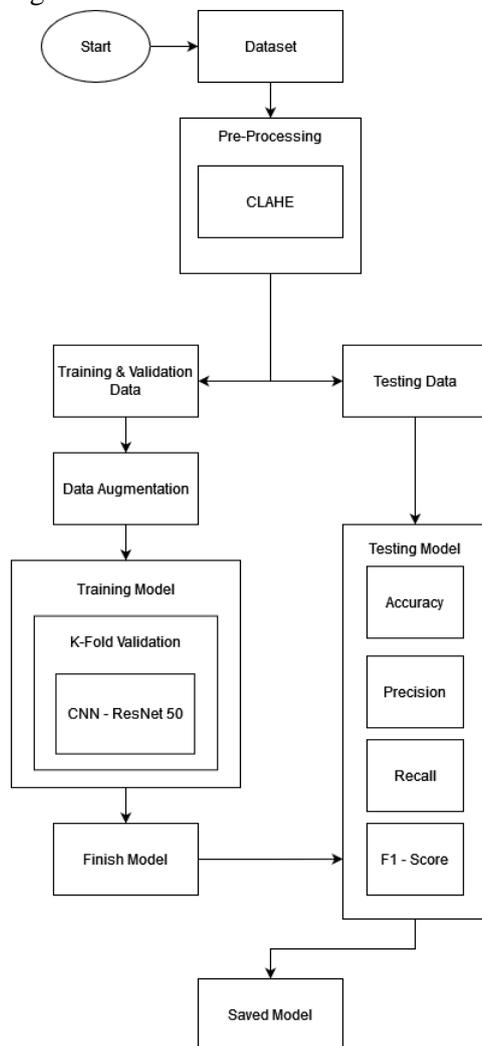
Akurasi merupakan tingkat rekognisi suatu label untuk mengenali label dan juga seberapa baik pengklasifikasian pada model untuk mengenali berbagai kelas, sedangkan presisi adalah tolak ukur ketepatan model, kemudian *recall* adalah sensitivitas suatu model atau *true positive rate*, dan *f measure* adalah *harmonic mean* dari presisi dan *recall* [27].

3. Hasil dan Pembahasan (Results and Discussions)

Penelitian ini akan menggunakan penelitian sebelum yang nantinya akan digunakan sebagai pembandingan untuk efektifitas model. Penelitian yang digunakan adalah “Identifikasi Batu Ginjal Menggunakan Extreme Learning Machine”, terdapat pembeda pada penelitian ini, terdapat proses augmentasi data sebelum dilakukan *training*, kemudian tidak ada proses segmentasi, dan metode *machine learning* yang digunakan adalah *convolutional neural network*.

3.1. Proses Pembuatan Model

Perancangan model *convolutional neural network* akan menjelaskan tentang bagaimana proses model tersebut dibuat, dalam proses ini nantinya akan menjelaskan *dataset* yang digunakan, proses *pre-processing*, *training* dan juga testing dari model.



Gambar 8 Proses Pembuatan Model Convolutional Neural Network

Pada struktur diatas akan terlihat perbedaan dengan penelitian sebelumnya, pada penelitian sebelumnya setelah *pre-processing*, ia melakukan segmentasi, sedangkan pada proses ini tidak terdapat, hal tersebut tetapi pada penelitian ini terdapat augmentasi. Kemudian pada proses training, penelitian ini menggunakan metode *convolutional neural network* dan *K-Fold validation* untuk mengukur performa model, sedangkan pada penelitian sebelumnya menggunakan *extreme learning*

machine saja dan tidak ada metode pengukuran yang digunakan untuk mengukur performa model sebelum *performance evaluation* menggunakan *confusion matrix*.

Pada proses *pre-processing* dataset yang telah didapatkan akan menjalani proses CLAHE yang dimana parameter yang digunakan adalah clip limit sebesar 8.0, setelah proses tersebut selesai dataset tersebut akan dengan komposisi, 60% *Training*, 20% *Validasi*, dan 20% *Testing*. Kemudian proses Augmentasi akan diterapkan pada data training dan data validasi untuk memperbanyak dataset, metode yang diterapkan pada augmentasi ini adalah image manipulation, berupa padding dengan parameter pad berukuran 32 dan flipping secara horizontal.

Pada proses training kita akan menggunakan, *convolutional neural network* dengan arsitektur ResNet 50 dan *stratified k-fold cross validation*. *Cross Validation* merupakan metode yang digunakan untuk mengukur kemampuan model menggunakan data yang tidak pernah ia lihat, artinya, menggunakan sampel terbatas untuk mengestimasi bagaimana model diharapkan bekerja saat digunakan untuk membuat prediksi pada data yang tidak digunakan selama proses *training* model. *Stratified k-Fold Cross Validation* adalah variasi dari *k-Fold* reguler tetapi ia tidak melakukan pemisahan pada dataset secara acak tetapi ia menggunakan rasio antar kelas, metode ini sangat menguntungkan jika digunakan pada saat kita memiliki data yang tidak seimbang.

3.2. Model Terbaik

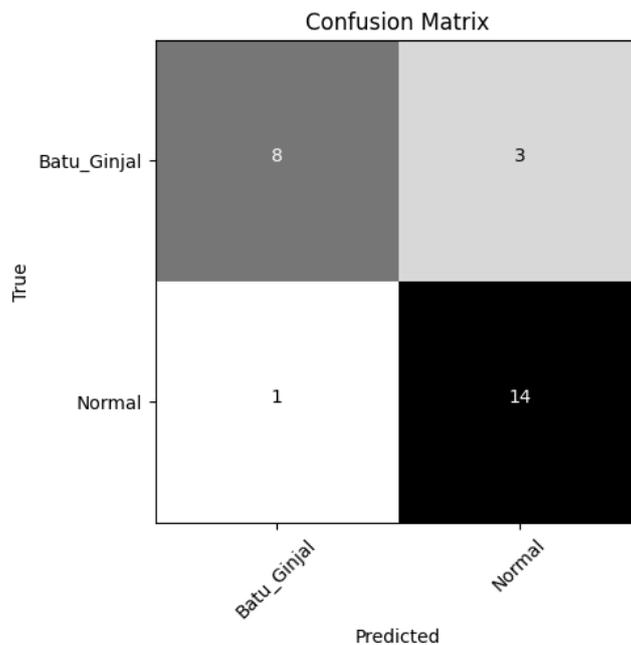
Pada saat proses *training* dilakukan, peneliti mencari model terbaik untuk digunakan dalam proses *training*, untuk mencari hal tersebut, dilakukanlah training menggunakan *epoch* berkelipatan 5, untuk mengukur apakah model tersebut adalah model terbaik maka setelah proses training selesai akan dilakukan proses *stratified k-fold cross validation* seperti yang dijelaskan pada bab sebelumnya. Setelah melakukan *stratified k-fold cross validation* menggunakan *epoch* berkelipatan 5, ternyata 35 yang menghasilkan model yang paling baik, berikut ini tabel perbandingan *epoch*.

Table 1 *Epoch Training*

| Epoch | Average Accuracy | Average Loss |
|-------|------------------|--------------|
| 5 | 0.58 | 0.70 |
| 10 | 0.62 | 0.66 |
| 15 | 0.67 | 0.61 |
| 20 | 0.73 | 0.55 |
| 25 | 0.77 | 0.48 |
| 30 | 0.81 | 0.36 |
| 35 | 0.85 | 0.25 |
| 40 | 0.77 | 0.29 |
| 45 | 0.72 | 0.35 |
| 50 | 0.66 | 0.44 |

Epoch 35 dipilih karena setelah *epoch* tersebut model mengalami penurunan performa, hal tersebut dilihat pada saat proses *cross validation* dilakukan. Setelah proses *cross validation* berjalan model menunjukkan penurunan akurasi, serta tingkat *loss* yang meningkat. Hal tersebut mengindikasikan bahwa model mengalami *overfitting*.

Setelah menemukan model terbaik maka, model tersebut akan melalu proses *testing*. Proses testing ini menggunakan dataset *testing* yang telah melalui proses CLAHE. Berikut ini adalah *confusion matrix* dari model tersebut.



Gambar 9 Confusion Matrix

Setelah mendapat confusion matrix kita dapat menghitung akurasi, presisi, *recall*, dan *F1-score*. Berikut ini adalah hasil dari pengolahan *confusion matrix*.

Table 2 Model Performance

| | Precision | Recall | F1-Score |
|----------------|-----------|--------|----------|
| Batu Ginjal | 0.89 | 0.73 | 0.80 |
| Normal | 0.82 | 0.93 | 0.87 |
| Accuracy Model | | | 0.85 |

Setelah mendapatkan hasil dari performa model, jika dibandingkan pada penelitian sebelumnya yang dilakukan menggunakan extreme machine learning [9]. Model yang menggunakan convolutional neural network memiliki performa lebih baik. Selain itu juga proses augmentasi, yang tidak ada pada penelitian sebelumnya ternyata pada penelitian ini juga sangat membantu dalam meningkatkan akurasi, karena jumlah dataset yang sedikit. Pada penelitian sebelumnya performa yang didapatkan adalah 80.76% sedangkan model ini bisa mendapatkan 85%.

3.3. Kesimpulan (Conclusion)

Tingkat performa dari model lebih baik dibandingkan dengan penelitian sebelumnya, adanya proses augmentasi sangat membantu dalam meningkatkan akurasi karena jumlah dataset yang sedikit, kemudian pada penelitian sebelumnya yang menggunakan metode extreme machine learning, total akurasi yang didapat adalah 80.76%. Sedangkan menggunakan metode convolutional neural network mendapatkan performa sebesar 85%. Selain itu, dataset yang digunakan untuk penelitian masih terlalu sedikit, diperlukan data yang lebih banyak untuk membuat model yang memiliki performa lebih bagus, kemudian untuk penelitian selanjutnya dapat menggunakan metode lain, atau dapat melakukan penambahan tahapan pada *pre-processing* untuk mendapatkan model yang lebih baik.

Ucapan Terima Kasih (Acknowledgement)

Penulis menyadari bahwa masih terdapat banyak kekurangan maka dari itu, penulis mengharapkan kepada pembaca untuk memberikan kritik dan saran yang bersifat konstruktif sehingga dapat menyempurnakan tugas akhir ini. Akhir kata, semoga tulisan ini dapat bermanfaat bagi banyak pihak. Serta penulis mengucapkan terima kasih kepada :

- Bapak Ardian Yusuf W., S.Kom., M.Kom & Mohammad Sholik., S.Kom., M.Kom. selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran dalam membimbing & mengarahkan penulis dalam penyusunan skripsi.
- Bapak Fidi Wincoko Putro, S.ST., M.Kom selaku kepala program studi Rekayasa Perangkat Lunak yang telah menyediakan waktunya dan tenaga dalam mengarahkan kami angkatan 2019.
- Kak Lisa Felicia S, yang telah membantu saya dengan memberikan data penelitian mengenai topik yang berkaitan dengan penelitian ini.
- Seluruh dosen program studi Rekayasa Perangkat Lunak atas segala ilmu, bimbingan, dan bantuan administratif selama penulis berkuliah.
- Seluruh teman-teman program studi Rekayasa Perangkat Lunak angkatan 2019 atas semangat, bantuan dan kebersamaan selama menjalani perkuliahan.

Daftar Pustaka

Aggarwal, C. C., 2018. *Neural Networks and Deep Learning*. Cham: Springer.

Han, J., Kamber, M. & Pei, J., 2012. *Data Mining : Concepts and Techniques 3rd Edition*. Massachusetts: Morgan Kaufmann.

Heaton, J., 2015. *Artificial Intelligence for Humans, Volume 3: Deep Learning and Neural Networks*. Washington DC: Heaton Research Inc.

Illah, P. P., Fauzi, H. & Siadari, T. S., 2022. Klasifikasi Penyakit Pneumonia Dan Covid-19 Berbasis Citra X-Ray Menggunakan Arsitektur Deep Residual Network. *eProceedings of Engineering*, 9(4), pp. 1837-1843.

Imron, A., 2019. *Analisis Sentimen Terhadap Tempat Wisata di Kabupaten Rembang Menggunakan Metode Naive Bayes Classifier*. Yogyakarta: Universitas Islam Indonesia.

Irawan, G. C., 2010. *Image Resizing Menggunakan Algoritma Seam Carving & Metode Reduksi Noise Median Filter*. Malang: Universitas Brawijaya.

Nashrullah, F., Wibowo, S. A. & Budiman, G., 2020. Investigasi Parameter Epoch Pada Arsitektur ResNet-50 Untuk Klasifikasi Pornografi. *Journal of Computer, Electronic, and Telecommunication*, 1(1), pp. 1-8.

O'Shea, K. T. & Nash, R., 2015. *An Introduction to Convolutional Neural Networks*. arXiv.

Putra, E. & Suartika, W., 2016. Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101. *Jurnal Teknik ITS*, 5(1), pp. 65-69.

Reza, A. M., 2004. Realization of the Contrast Limited Adaptive Histogram Equalization (CLAHE) for Real-Time Image Enhancement. *The Journal of VLSI Signal Processing-Systems for Signal*, Volume 38, pp. 35-44.

Silaen, L. F., 2021. *Identifikasi Batu Ginjal Menggunakan Extreme Learning Machine*. Medan: Universitas Sumatra Utara.

Singh, P., Mukundan, R. & Ryke, R. D., 2020. Feature Enhancement in Medical Ultrasound Videos Using Contrast-Limited Adaptive Histogram Equalization. *Journal of digital imaging*, 33(1), pp. 273-285.

Sonka, M., Hlavac, V. & Boyle, R., 2014. *Image Processing, Analysis, and Machine Vision* 4th Edition. Massachusetts: Cengage Learning.

Universitas Airlangga News, 2020. Mengenal Profil Pasien Batu Saluran Kemih. [Online] Available at: <https://news.unair.ac.id/2020/04/07/mengenal-profil-pasien-batu-saluran-kemih/>

Yadav, G., Maheshwari, S. & Agarwal, A., 2014. *Contrast Limited Adaptive Histogram Equalization Based Enhancement For Real Time Video System*. Delhi, s.n.

Yang, S. et al., 2022. *Image Data Augmentation for Deep Learning: A Survey*. arXiv.