

KLASIFIKASI KUALITAS PEWARNAAN ZIEHL NEELSEN PADA SAMPEL DAHAK TBC DENGAN MENGGUNAKAN MACHINE LEARNING

1st Faris Naufal Arif

School of Electrical Engineering

Telkom University

Bandung, Indonesia

farissun@student.telkomuniversity.ac.id

2nd Dilazapira

School of Electrical Engineering

Telkom University

Bandung, Indonesia

dilazapira@telkomuniversity.ac.id

3rd Siti Sadira Muntaz Dinar

School of Electrical Engineering

Telkom University

Bandung, Indonesia

sadiramuntaz@telkomuniversity.ac.id

4th Inung Wijayanto

School of Electrical Engineering

Telkom University

Bandung, Indonesia

iwijayanto@telkomuniversity.ac.id

5th Suci Aulia

School of Applied Sciences

Telkom University

Bandung, Indonesia

suciaulia@telkomuniversity.ac.id

Abstrak—Penyakit Tuberkulosis (TBC) masih menjadi masalah kesehatan global yang memerlukan diagnosis yang cepat dan akurat. Salah satu metode diagnosis yang umum digunakan adalah pemeriksaan mikroskopis dengan pewarnaan Ziehl-Neelsen (ZN) pada sampel dahak. Namun, kualitas pewarnaan yang kurang optimal seperti *Over Staining* atau *Less Staining* dapat memengaruhi akurasi pembacaan mikroskopis dan menyebabkan kesalahan diagnosis. Untuk mengatasi masalah tersebut, penelitian ini mengembangkan sebuah aplikasi *standalone* berbasis ML yang dapat melakukan klasifikasi kualitas pewarnaan ZN dan menghitung jumlah bakteri TBC secara otomatis. Sistem ini dibangun agar dapat berjalan secara lokal (*offline*) dan terintegrasi antara model klasifikasi berbasis CNN dan deteksi bakteri berbasis YOLOv11. Berdasarkan hasil pengujian terhadap *dataset* uji, sistem klasifikasi dengan model CNN mencapai akurasi, presisi, sensitivitas, spesifisitas, dan F1-skor sebesar 100% pada setiap kelas. Pada tahap deteksi bakteri, model YOLOv11 menunjukkan performa evaluasi dengan mAP, *precision*, *recall*, dan F1-skor lebih dari 80%. Waktu rata-rata pemrosesan gambar pada 30 laptop berbeda tercatat kurang dari 3 detik per gambar. Hasil ini menunjukkan bahwa sistem yang dikembangkan memiliki performa tinggi, fleksibel, dan siap diimplementasikan dalam skala operasional laboratorium maupun daerah dengan keterbatasan jaringan.

Kata Kunci—Aplikasi standalone, Klasifikasi gambar, Machine Learning, Tuberkulosis, Ziehl Neelsen.

I. PENDAHULUAN

Tuberculosis atau biasa dikenal dengan singkatan TBC merupakan penyakit menular yang disebabkan oleh bakteri *Mycobacterium tuberculosis* (MTBC) [1]. Bakteri ini menyebar ketika pasien yang sudah terjangkit penyakit TBC mengeluarkan bakteri ke udara seperti batuk, bersin, atau membuang air liur sembarangan [2]. Menurut *World Health Organization* (WHO) Report 2023, Indonesia menempati peringkat dua di dunia dengan jumlah kasus TBC terbanyak

setelah India [2]. Kasus penyakit TBC di Indonesia mencapai 800 ribu pasien dan kasus kematian mencapai 23 ribu pasien yang meninggal dikarenakan TBC [1].

Pasien yang terjangkit bakteri TBC perlu waktu 6 bulan untuk tahap penyembuhan. Terdapat dua kemungkinan yang terjadi saat pasien berada pada tahap penyembuhan, yaitu sembuh atau *Multidrug-Resistant Tuberculosis* (MDR-TBC) [3]. MDR-TBC merupakan kondisi dimana pasien kebal terhadap antibiotik yang diberikan, sehingga dokter memberikan tingkat antibiotik yang lebih tinggi untuk diberikan kepada pasien [4]. MDR-TBC terjadi ketika pasien tidak menyelesaikan pengobatan sesuai dosis dan durasi yang ditentukan atau tertular oleh pasien yang memiliki bakteri TBC yang resisten terhadap antibiotik pada dosis yang diberikan [5].

Pada pengobatan TBC, screening secara sistematis sangat diperlukan untuk mendapatkan sampel. Hal tersebut dilakukan dengan menggunakan tes, pemeriksaan, atau prosedur lain yang dapat diterapkan dengan cepat [6]. Beberapa jenis screening utama yang dilakukan pada pengobatan TBC meliputi *Interferon-Gamma Release Assays* (IGRA), *Rontgen*, *Polymerase Chain Reaction* (PCR), dan konvensional mikroskopis.

Metode yang cocok diterapkan di Indonesia untuk diagnosis TBC adalah mikroskopis sputum smear, karena metode ini merupakan salah satu yang paling efektif dalam mengidentifikasi orang dengan TBC menular [7]. Selain itu, metode ini tetap menjadi pilihan utama untuk diagnosis TBC di negara-negara berpenghasilan rendah dan menengah atau *Low and Middle Income Country* (LMIC). Mikroskopis sputum smear tidak hanya terbukti efektif secara biaya

dalam mendiagnosis pasien dengan TBC menular, tetapi juga dalam memantau perkembangan pengobatan mereka serta mengonfirmasi kesembuhan [8]. Pewarnaan yang digunakan untuk metode mikroskopis sputum smear adalah *Ziehl Neelsen* (ZN) karena biayanya yang rendah dan kemudahan dalam pemeliharaan peralatan, menjadikannya pilihan yang tepat dalam kondisi terbatas [9]. Penggunaan Pewarnaan ZN menargetkan MTBC yang setelah diberi pewarnaan, bakteri akan tampak merah dikarenakan MTBC merupakan bakteri yang tahan asam (*acid-fast bacilli*) sehingga mudah dibedakan dari struktur seluler lainnya yang berlatar belakang biru [10].

Salah satu faktor yang menghambat program pemberantasan TBC adalah belum adanya diagnosis cepat dan tepat untuk menentukan bakteri MTBC dalam sputum [11]. Diagnosis cepat tepat tersebut sangatlah penting bukan untuk menentukan pengobatannya saja tetapi untuk memutus rantai penularan [12]. Teknik diagnostik yang sering melibatkan analisis mikroskopis adalah *GeneXpert* dengan pewarnaan ZN. Metode pewarnaan ZN mempunyai keuntungan untuk analisis penyakit TBC, yaitu hasil dari mikroskopis yang berlatar belakang biru cerah yang membuat basil merah menonjol dan juga biaya dari reagen yang relatif rendah [13]. Metode pewarnaan ZN memiliki beberapa keterbatasan dalam interpretasi hasil sampel tersebut. Selain itu prosesnya bisa memakan waktu hampir sebulan atau bahkan lebih lama [14]. Berbagai faktor dapat memengaruhi hasil pemeriksaan mikroskopis, seperti keterampilan dan ketelitian petugas laboratorium yang kurang memadai, serta jenis mikroskop yang digunakan [15]. Oleh sebab itu Tujuan dari penelitian ini adalah untuk memungkinkan proses analisis yang lebih cepat dan akurat, sehingga dapat mengurangi kesalahan manusia dan meningkatkan akurasi, efisiensi dan konsistensi dalam mendeteksi dan menghitung jumlah bakteri TBC.

II. KAJIAN TEORI

A. Convolutional Neural Network (CNN)

Secara teknis, CNN merupakan suatu arsitektur yang dapat dilatih yang terdiri dari tiga tahap, yaitu tahap masukan (input), tahap pemrosesan, dan tahap keluaran (output). CNN merupakan salah satu arsitektur dalam deep learning yang direkomendasikan dalam mengidentifikasi pola spasial. CNN mampu mendeteksi dan mengenali objek yang terdapat pada sebuah gambar [16]. Tahap masukan pada CNN merupakan tahap untuk menerima data mentah, seperti gambar atau video yang dapat diolah melalui beberapa lapisan, seperti convolutional layers, pooling layers, dan fully connected layers. Setiap lapisan memiliki peran yang beragam, seperti mengekstraksi fitur, mengurangi dimensi data, dan menghasilkan prediksi [17]. CNN mampu mendeteksi dan mengenali objek yang terdapat pada sebuah gambar, seperti pola, tekstur, tepi atau bentuk tertentu. Arsitektur ini sangat cocok untuk deteksi objek, segmentasi citra, dan klasifikasi gambar.

Terdapat beberapa tahapan CNN yang meliputi operasi konvolusi dan penggabungan lapisan untuk memproses data.

Terdapat 4 tahapan pada arsitektur CNN yang meliputi tahap masukan (input), tahap ekstraksi fitur, tahap klasifikasi, dan tahap keluaran (output) [17].

B. Confusion Matrik

Confusion matrix merupakan alat yang sering digunakan untuk mengevaluasi kinerja model klasifikasi dalam ML. Matriks ini berbentuk tabel yang berisikan jumlah prediksi yang benar dan salah yang dibuat oleh model untuk setiap kelas dalam data uji. Pada *Confusion Matrix*, terdiri dari 4 komponen yaitu true positive (TP), true negative (TN), false positive (FP), dan false negative (FN) [18]. TP merupakan jumlah data yang benar diklasifikasikan ke kelas positif. TN merupakan jumlah data yang benar diklasifikasikan ke kelas negatif. FP merupakan jumlah data yang salah diklasifikasikan ke kelas positif. FN merupakan jumlah data yang salah di klasifikasikan ke kelas negatif.

Tabel I: Tabel Confusion Matrix

Kelas Aktual	Kelas Prediksi	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Tabel I merupakan tabel posisi pada setiap komponen *confusion matrik*. Dengan 4 komponen tersebut dapat mengevaluasi aspek-aspek spesifik dari kinerja model, seperti akurasi, presisi, *recall* (Sensitivitas), spesifisitas, dan F1-Skor. Berikut merupakan penjelasan mengenai Akurasi, Precision, Recall (Sensitivitas), Specificity, dan F1-skor:

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (4)$$

$$\text{F1-skor} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

C. YOLOv11

Computer Vision (CV) adalah sebuah konsep yang memungkinkan mesin untuk memahami data visual. Salah satu aspek dari CV adalah deteksi objek [19]. Deteksi objek adalah pelibatan identifikasi dan lokalisasi objek yang melibatkan identifikasi dan lokalisasi objek secara tepat di dalam gambar atau video [20]. Kemajuan signifikan dalam deteksi objek dicapai dengan diperkenalkannya algoritma *You Only Look Once* (YOLO) oleh Redmon dkk. pada tahun 2015 [21]. Metodologi YOLO berbeda dari proses deteksi dua langkah tradisional dengan memperlakukan deteksi objek

sebagai masalah regresi [21]. Metode ini menggunakan jaringan *convolutional network* tunggal untuk memprediksi batas bidang dan probabilitas kelas secara simultan di seluruh gambar, sehingga meningkatkan proses deteksi dibandingkan dengan metode tradisional yang lebih kompleks [22]. YOLOv11 adalah versi terbaru dari seri terbaru dari YOLO yang diluncurkan pada tahun 2024. YOLOv11 memberikan peningkatan substansi dalam arsitektur maupun metode pelatihan, meningkatkan akurasi, kecepatan, dan efisiensi [23].

YOLO merevolusi deteksi objek dengan memperkenalkan arsitektur neural network yang secara bersamaan mendukung regresi kotak pembatas (*bounding box*) dan tugas klasifikasi objek [24]. Pendekatan terpadu ini merupakan perubahan signifikan dari metode deteksi dua tahap tradisional. Arsitektur YOLO terdiri dari tiga komponen utama yaitu *backbone* sebagai ekstraktor fitur utama, menggunakan jaringan *convolutional network* untuk mengubah data citra mentah menjadi *multi scale feature maps*. Kedua adalah komponen *neck* yang berfungsi perantara, memanfaatkan lapisan khusus untuk mengagregasi dan meningkatkan representasi fitur di berbagai skala. Ketiga adalah *head* yang merupakan komponen yang berfungsi sebagai mekanisme prediksi, menghasilkan output untuk objek lokalisasi dan klasifikasi objek berdasarkan *feature maps* yang sudah disempurnakan [23].

D. Roboflow

Roboflow adalah *platform* lengkap yang mendukung seluruh tahapan pengembangan model *computer vision*, mulai dari pelabelan data, pra-pemrosesan, hingga pelatihan. Platform ini mendukung berbagai format gambar umum dan menyediakan augmentasi data untuk meningkatkan performa model. Keunggulan Roboflow terletak pada kemampuannya terintegrasi dengan *framework* populer seperti TensorFlow, PyTorch, dan YOLO, sehingga model dapat dijalankan di berbagai perangkat keras. Roboflow juga memudahkan pengelolaan dan pembersihan dataset, serta menyediakan proses deployment yang praktis untuk penggunaan langsung di dunia nyata [25]

E. Random Search

Random Search adalah metode untuk memilih nilai-nilai hyperparameter secara acak dan mengevaluasi kinerjanya di setiap iterasi [26]. Setelah beberapa percobaan, kombinasi dengan performa terbaik akan dipilih. Metode ini cocok digunakan ketika jumlah hyperparameter banyak dan ruang pencariannya luas.

F. Pretrained model

Model pra-trained adalah pendekatan yang menggunakan model yang sudah dilatih sebelumnya sebagai titik awal untuk menyelesaikan masalah baru. Dengan cara ini, parameter model hanya perlu diperbarui (*fine-tuned*) agar sesuai dengan tugas spesifik, sehingga proses pelatihan menjadi lebih cepat dan efisien dibanding melatih model dari awal [27].

G. Adam Optimizer

Adam (Adaptive Moment Estimation) adalah algoritma optimasi yang menggabungkan keunggulan RMSProp dan Momentum [28]. Adam optimizer secara adaptif akan menyesuaikan *learning rate* untuk setiap parameter model. Adam merupakan salah satu optimizer yang banyak digunakan dalam *deep learning* karena efisiensi pada dataset besar dan parameter kompleks [28]. Pada proses pelatihan, Adam secara dinamis mengubah *learning rate* berdasarkan kondisi selama pelatihan, sehingga model dapat lebih efektif mencapai hasil yang optimal tanpa memerlukan penyesuaian manual yang rumit.

H. Visual Studio Code

Visual Studio Code (VSCode) adalah sebuah teks editor handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform. Multiplatform artinya tersedia juga untuk versi Linux, Mac, dan Windows atau bisa dipakai di banyak OS [29]. VSCode dikembangkan oleh Microsoft pertama kali di tanggal 29 April 2015. VSCode mendukung bermacam-macam bahasa pemrograman seperti C, C++, Java, Python, dan banyak lagi [29]. Visual Studio code memiliki banyak sekali fitur-fitur seperti Git Integration, Debugging, dan fitur ekstensi [30].

III. METODE

A. Spesifikasi Sistem

Untuk memastikan aplikasi TBClassify yang dirancang dapat memenuhi standar yang ditetapkan, diperlukan pendekatan yang cermat dalam pengukuran dan spesifikasi. Tujuan dari spesifikasi sistem adalah untuk menetapkan batasan yang diperlukan oleh sistem guna mencapai hasil yang optimal. Berikut merupakan tabel batasan dan spesifikasi utama yang telah ditetapkan untuk mengukur keberhasilan solusi yang dirancang.

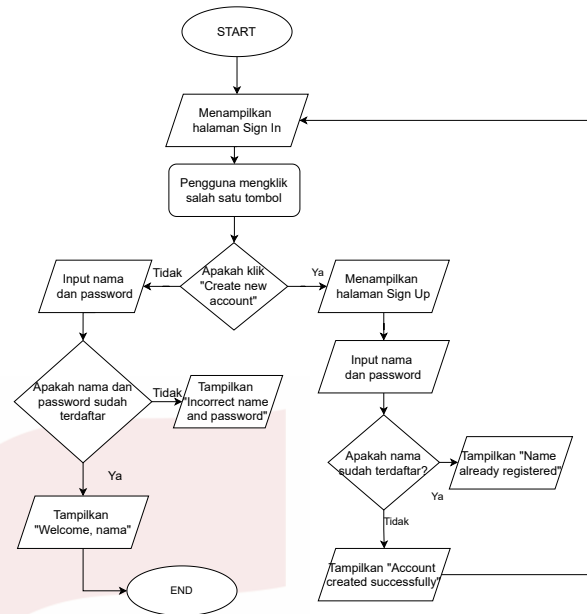
Berdasarkan Tabel II, spesifikasi dan batasan yang telah ditetapkan pada aplikasi TBClassify adalah dasar untuk mengukur keberhasilan dalam pengembangan aplikasi TBClassify. Spesifikasi tersebut menjadi acuan utama dalam mengukur keberhasilan pengembangan aplikasi, mulai dari akurasi klasifikasi, deteksi bakteri, validasi logika sistem, hingga kepuasan pengguna. Secara keseluruhan, batasan yang telah ditentukan memberikan kerangka kerja yang jelas dalam mengarahkan desain dan implementasi program, memastikan sistem tetap fokus pada ruang lingkup yang relevan dan realistis. Maka dari itu, aplikasi TBClassify diharapkan tidak hanya memenuhi kebutuhan teknis, tetapi juga sejalan dengan standar ilmiah, legal, dan etika yang berlaku.

B. Desain Sistem

Aplikasi TBClassify adalah pengembangan aplikasi standalone yang dapat dioperasikan secara lokal tanpa ketergantungan pada koneksi internet. Aplikasi ini dirancang khusus untuk klasifikasi kualitas pewarnaan ZN serta dapat mendeteksi dan menghitung bakteri TBC pada gambar menggunakan teknologi ML. Pemrosesan dilakukan secara

Tabel II: Spesifikasi Sistem

Spesifikasi	Keterangan
Klasifikasi kualitas pewarnaan ZN	Mengukur akurasi klasifikasi dan konsistensi model CNN terhadap variasi kualitas pewarnaan ZN (<i>Good Staining, Less Staining, Over Staining</i>) menggunakan dataset uji berlabel, dengan target semua metrik (akurasi, recall, specificity, precision, F1-skor) lebih dari 90%
Deteksi dan Perhitungan Bakteri	Mengukur presisi <i>bounding box</i> , kemampuan menghitung jumlah bakteri TBC, dan ketahanan terhadap <i>false positive</i> hanya pada gambar dengan hasil klasifikasi <i>Good Staining</i> , menggunakan mAP dan metrik evaluasi lainnya dengan target lebih dari 80%
Validasi Logika Sistem	Memastikan integrasi model CNN dan YOLO berjalan sesuai alur kerja, mulai dari Login, Input gambar, klasifikasi, deteksi, dengan logika yang benar, termasuk notifikasi untuk <i>Less Staining/Over Staining</i> dan eksekusi deteksi hanya pada <i>Good Staining</i>
Waktu Respon Sistem	Sistem mampu memproses klasifikasi dan deteksi bakteri TBC dengan waktu respon total kurang dari 3 detik pada perangkat uji
Kepuasan Pengguna	Berdasarkan kuisioner (skala 1–5), mengukur kemudahan penggunaan antarmuka dan keandalan sistem, dengan fokus pada konsistensi hasil dan kenyamanan pengguna



Gambar 1: Flowchart Registrasi Pengguna

langsung pada perangkat pengguna, sehingga sistem dapat digunakan di laboratorium atau fasilitas kesehatan dengan keterbatasan konektivitas jaringan. Model CNN digunakan untuk klasifikasi kualitas pewarnaan ZN ke dalam tiga kategori, sedangkan model YOLO digunakan untuk mendeteksi dan menghitung bakteri TBC. Berikut merupakan alur kerja aplikasi TBClassify dengan desain sebagai berikut.

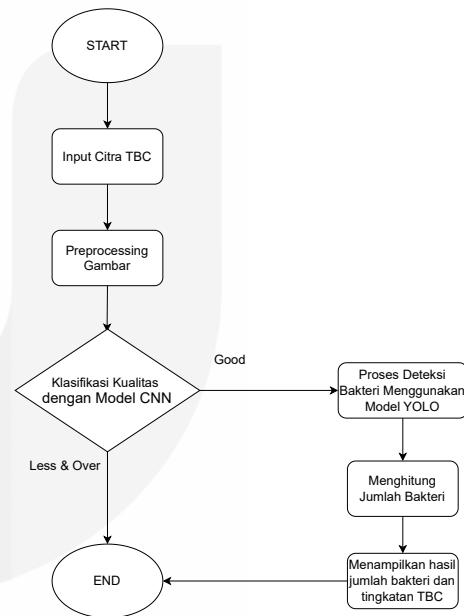
1) *Flowchart*: Flowchart digunakan untuk memvisualisasikan proses kerja aplikasi standalone mulai dari penerimaan input citra hingga penyajian hasil analisis kepada pengguna. Flowchart dibagi menjadi dua, yaitu flowchart registrasi dan proses analisis. Gambar 1 adalah flowchart registrasi dan proses analisis dan Gambar 2 adalah Flowchart proses analisis input citra

2) *Desain Aplikasi*: Desain aplikasi berperan penting untuk memastikan bahwa seluruh komponen mulai dari input data, pemrosesan, hingga penyajian hasil analisis dapat diintegrasikan dengan baik dan dioperasikan secara efisien. Dengan desain aplikasi yang matang, pengguna mendapatkan kenyamanan dan kemudahan untuk dapat menggunakan aplikasi dengan baik. Gambar 3 adalah desain dari aplikasi yang kami buat.

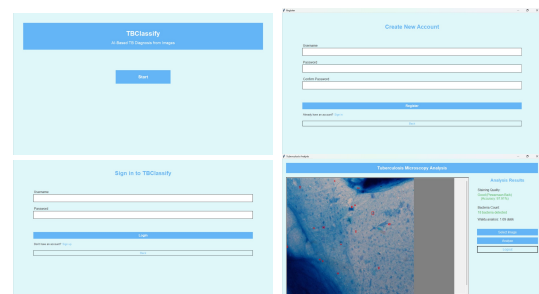
C. Metode Pengukuran yang Sesuai

Setelah menentukan batasan dan spesifikasi, pengukuran solusi diperlukan untuk memastikan solusi memenuhi spesifikasi yang ditetapkan. Tujuan dilakukan pengukuran solusi adalah untuk menetapkan batasan yang diperlukan oleh sistem guna mencapai hasil yang optimal. Berikut merupakan rincian mengenai proses pengukuran dari setiap spesifikasi.

1) *Klasifikasi Kualitas Pewarnaan ZN*: Pengukuran ini bertujuan untuk memvalidasi kemampuan model CNN da-



Gambar 2: Flowchart Proses Analisis Input Citra



Gambar 3: Desain Aplikasi

lam mengklasifikasikan kualitas pewarnaan ZN pada sampel dahak ke dalam tiga kategori: *Good Staining*, *Less Staining*, dan *Over Staining*. Nilai reliabilitas pengujian modul ini diukur dari akurasi klasifikasi dan konsistensi model terhadap variasi kualitas pewarnaan ZN pada citra yang diuji.

Tabel III: Pengukuran Klasifikasi Kualitas Pewarnaan ZN

Hal	Keterangan
Rincian yang Diukur	Mengukur akurasi klasifikasi dan konsistensi model terhadap kualitas gambar [31]
Metode Pengukuran	Evaluasi menggunakan dataset uji yang sudah dilabeli menjadi tiga kategori
Mekanisme Pengukuran	Menggunakan dataset uji yang tidak digunakan selama proses pelatihan dan menghitung metrik evaluasi mulai dari akurasi, recall, specificity, precision, F1-skor menggunakan <i>confusion matrix</i> [17], [22] dan semua metrik evaluasi mendapatkan nilai minimal 90% pada setiap kategori klasifikasi

Tabel III merupakan mekanisme pengukuran klasifikasi kualitas pewarnaan ZN dengan confusion matrix. Dalam pengujian ini, model yang sudah dilatih diuji menggunakan dataset uji yang belum pernah digunakan selama proses pelatihan. Proses pengukuran menggunakan beberapa metrik evaluasi, yaitu akurasi, recall, specificity, precision, dan F1-skor. Agar sistem dikatakan berhasil, seluruh nilai metrik evaluasi pada setiap kategori klasifikasi mendapatkan nilai minimal 90

2) *Deteksi dan Perhitungan Bakteri TBC* : Pengujian ini mengevaluasi performa model YOLO dalam mendeteksi dan menghitung bakteri TBC pada sampel yang telah di-klasifikasikan sebagai *Good Staining*. Nilai reliabilitas yang diuji meliputi mAP, *recall*, *specificity*, *precision*, dan F1-skor. Berikut merupakan tabel mekanisme pengukuran deteksi dan perhitungan bakteri TBC.

Tabel IV: Pengukuran Deteksi dan Perhitungan Bakteri TBC

Hal	Keterangan
Rincian yang Diukur	Aspek yang diuji meliputi akurasi deteksi (<i>precision bounding box</i>) [21], kemampuan menghitung jumlah bakteri, serta ketahanan terhadap <i>false positive</i> (deteksi non-bakteri) [17].
Metode Pengukuran	Menggunakan dataset terpisah dengan anotasi <i>bounding box</i> bakteri TBC [31] dan hanya dijalankan ketika hasil klasifikasi memprediksi <i>Good Staining</i> .
Mekanisme Pengukuran	Menggunakan dataset uji yang tidak digunakan selama proses pelatihan dan menghitung metrik evaluasi mulai dari mAP, recall, specificity, precision, F1-skor menggunakan <i>confusion matrix</i> [17], [22] dan semua metrik evaluasi mendapatkan nilai minimal 80% pada setiap kategori klasifikasi.

Tabel IV merupakan mekanisme pengukuran klasifikasi kualitas pewarnaan ZN dengan confusion matrix. Pengujian menggunakan dataset uji yang telah dianotasi dengan bounding box yang berisi bakteri TBC dan tidak digunakan selama proses pelatihan. Pengukuran ini dilakukan dengan menghitung nilai mean average precision (mAP) serta metrik

evaluasi lainnya seperti recall, specificity, precision, F1-skor. Seluruh nilai metrik evaluasi minimal ditargetkan mencapai 80% pada setiap kelas bakteri yang diuji. Aspek yang juga diuji mencakup ketahanan sistem terhadap false positive (FP) atau deteksi keliru pada area yang bukan bakteri.

3) *Validasi Logika Sistem*: Pengujian validasi logika sistem dilakukan melalui simulasi alur kerja sistem secara menyeluruh, yang terdiri atas: proses sign in, input citra, klasifikasi kualitas pewarnaan ZN, deteksi dan perhitungan bakteri TBC (jika hasil klasifikasi adalah *Good Staining*), dan analisis hasil akhir. Berikut merupakan tabel mekanisme pengukuran validasi logika sistem.

Tabel V: Pengukuran Validasi Logika Sistem

Hal	Keterangan
Rincian yang Diukur	Validasi logika sistem (notifikasi untuk hasil <i>Less/Over Staining</i>) dan stabilitas integrasi antara komponen model CNN dengan YOLO dalam kondisi operasional yang nyata.
Metode Pengukuran	Simulasi alur kerja sistem: <i>Sign in</i> → login → input gambar → klasifikasi → deteksi bakteri (Jika <i>Good Staining</i>) → output analisis [31].
Mekanisme Pengukuran	Input citra <i>Less/Over Staining</i> , pastikan sistem menampilkan notifikasi tanpa menjalankan perhitungan bakteri [32]. Input citra <i>Good Staining</i> , pastikan sistem menjalankan deteksi dan menampilkan jumlah bakteri [17].

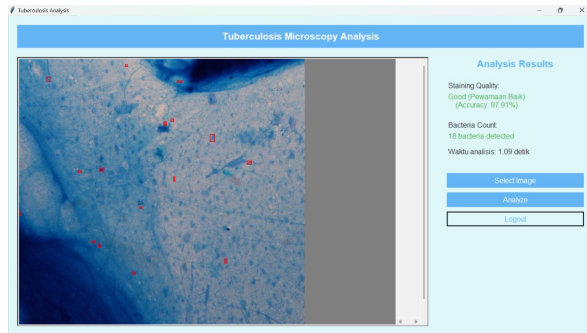
Tabel V merupakan mekanisme pengukuran validasi logika sistem menggunakan simulasi. Validasi logika diuji dengan dua skenario: pertama, apabila input citra termasuk kategori *Over Staining*, maka sistem harus menampilkan notifikasi pemberitahuan tanpa melanjutkan ke tahap deteksi bakteri TBC. Kedua, apabila input citra termasuk kategori *Good Staining*, maka sistem wajib menjalankan deteksi dan menampilkan hasil berupa jumlah bakteri TBC yang terdeteksi secara otomatis.

4) *Waktu Respon Sistem* : Waktu respon sistem dilakukan dengan menjalankan aplikasi pada 30 perangkat laptop atau pc yang berbeda untuk mengukur proses pada tahap klasifikasi hingga deteksi bakteri TBC. Berikut merupakan tabel mekanisme pengukuran waktu respon sistem.

Tabel VI: Pengukuran Waktu Respon Sistem

Hal	Keterangan
Rincian yang Diukur	Pengukuran waktu respon sistem untuk memastikan analisis berjalan dengan cepat.
Metode Pengukuran	Menjalankan aplikasi pada 30 perangkat laptop atau PC yang berbeda [23].
Mekanisme Pengukuran	Ketika menjalankan aplikasi, waktu inferensi per citra pada klasifikasi dan deteksi bakteri TBC kurang dari 3 detik [31].

Tabel VI merupakan mekanisme pengukuran waktu respon sistem dengan menjalankan aplikasi pada 30 laptop berbeda. Sistem dinyatakan memenuhi spesifikasi apabila total waktu inferensi per gambar (dari klasifikasi hingga deteksi) tidak melebihi 3 detik, sesuai target efisiensi minimum yang ditetapkan.



Gambar 4: Fitur Utama TBClassify

5) *Kepuasan Pengguna*: Pengujian sistem juga mencakup pengukuran kepuasan pengguna, yang diperoleh melalui penyebaran kuisioner dengan skala 1-5 yang merepresentasikan tingkat keandalan dan kenyamanan mereka menggunakan aplikasi. Berikut merupakan tabel mekanisme pengukuran kepuasan pengguna. Tabel VII merupakan mekanisme pe-

Tabel VII: Pengukuran Kepuasan Pengguna

Hal	Keterangan
Rincian yang Diukur	Uji kepuasan pengguna untuk menilai kemudahan dan keandalan sistem.
Metode Pengukuran	Pengisian kuisioner (Skala 1–5) terkait tingkat keandalan dan kenyamanan menggunakan aplikasi [25].
Mekanisme Pengukuran	Kuisioner berisi pertanyaan terkait keandalan sistem [33] dan kemudahan penggunaan antarmuka [20].

ngukuran kepuasan pengguna dengan penyebaran kuisioner dengan skala 1-5. Aspek yang dinilai meliputi keandalan sistem dalam memberikan hasil yang konsisten dan kemudahan penggunaan antarmuka pengguna. Tujuan dari pengukuran ini adalah memastikan bahwa sistem dalam mmeberikan hasil yang fungsional, tetapi juga diterima secara positif oleh pengguna dari sisi pengalaman penggunaan (*user experience*).

IV. HASIL DAN PEMBAHASAN

A. Implementasi TBClassify

TBClassify merupakan aplikasi yang dirancang untuk mendukung proses diagnosis penyakit TBC melalui analisis citra bakteri. Dalam aplikasi TBClassify terdiri dari beberapa komponen utama:

- Antarmuka Pengguna (GUI): Dibangun dengan menggunakan Tkinter dengan berbagai elemen yaitu Frame, Label, Button, Entry, dan Canvas
- Database lokal: Data pengguna disimpan dalam file JSON (users.json) dan data diproses langsung dari file lokal
- Model ML: Menggunakan 2 model yaitu model CNN yang digunakan untuk menentukan kualitas pewarnaan citra bakteri, dan model YOLO untuk mendeteksi bakteri pada citra bakteri.

TBClassify memiliki fitur utama pada halaman utama yaitu perhitungan dan deteksi pada citra mikroskopis seperti yang tertera di gambar 4. Proses dimulai dengan klasifikasi kualitas pewarnaan menggunakan model CNN yang telah dilatih untuk membedakan kualitas pewarnaan yang baik dan buruk. Hanya gambar yang memiliki pewarnaan 'Good' saja yang akan dilanjutkan ke tahap perhitungan bakteri, dan akan menampilkan jumlah dalam range serta posisi bakteri melalui *bounding box* berwarna merah

B. Implementasi Model CNN untuk Klasifikasi Kualitas Pewarnaan ZN

Model yang dipilih untuk sistem klasifikasi kualitas pewarnaan ZN adalah hasil dari optimasi terhadap model CNN sebelumnya. Dalam tahap pengembangan, Penulis melakukan optimasi dengan menggunakan *balancing dataset* dan menggunakan teknik *callback* untuk meningkatkan akurasi pada proses pelatihan.

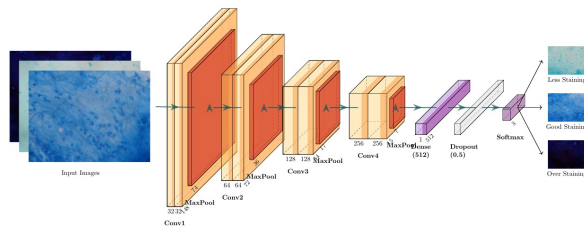
Dataset dengan kategori tiga kelas pewarnaan yang digunakan selama pelatihan model, yaitu *LessStaining* (40 sampel), *GoodStaining* (76 sampel), dan *OverStaining* (18 sampel). Dataset awal bersifat tidak seimbang, dengan kelas *GoodStaining* mendominasi dan kelas *OverStaining* memiliki jumlah sampel paling sedikit. Ketidakseimbangan ini menjadi perhatian utama sebelum memasuki tahap pelatihan model. Untuk mengatasinya, dilakukan pre-processing dataset dengan menerapkan tehnik *visual data augmentation* dan *SMOTE (Synthetic Minority Oversampling Technique)*. Hasilnya, setiap kelas memiliki jumlah sampel yang seimbang, yaitu 70 sampel, sehingga model tidak bias terhadap kelas mayoritas dan dapat belajar secara adil dari semua kelas.

Tabel VIII: Proses Optimasi Model CNN

Tahapan	Teknik Optimasi	Catatan
Model Awal	Dataset Asli Tanpa regulasi overfitting	Akurasi yang didapat 88%
Model Optimasi Pertama	Menyeimbangkan dataset dengan teknik augmentasi Menambahkan teknik <i>EarlyStopping</i> untuk target akurasi 95%	Akurasi meningkat menjadi 95%
Model Optimasi Kedua	Target akurasi dinaikkan menjadi 97%	Model stabil, Akurasi 100%

Tabel VIII menjelaskan bahwa model CNN dioptimalkan melalui tiga tahap utama, yaitu model awal, optimasi pertama, dan optimasi kedua. Proses optimasi difokuskan pada tiga aspek, yaitu menyeimbangkan dataset, menerapkan tehnik regulasi untuk mencegah overfitting seperti *EarlyStopping*, serta meningkatkan target akurasi hingga di atas 97%. Upaya ini menghasilkan model yang sangat akurat dan stabil tanpa menunjukkan tanda-tanda overfitting. Berikut merupakan rancangan arsitektur yang digunakan.

Gambar 5 menunjukkan parameter rancangan yang digunakan dalam model klasifikasi dengan jenis pooling Maximum Pooling, menggunakan kernel size 3x3 pada convolutional layer. Jumlah filter yang digunakan adalah 32, 64,



Gambar 5: Arsitektur Model CNN

128, dan 256, dengan proses konvolusi sesuai jumlah filter tersebut [28]. Kernel size merupakan dimensi kernel untuk proses konvolusi, sedangkan pooling layer berfungsi mengurangi ukuran spasial dengan memilih piksel terpenting. Maximum pooling memilih piksel dengan nilai terbesar pada area sebelumnya. Dense layer memiliki 512 neuron, dengan fungsi aktivasi ReLU. Output layer memiliki 3 neuron sesuai jumlah kelas. Optimizer yang digunakan adalah Adam [28]. Proses pelatihan menggunakan dataset pelatihan dengan proporsi 70% dari dataset dan validasi dengan proporsi 20% dari dataset.

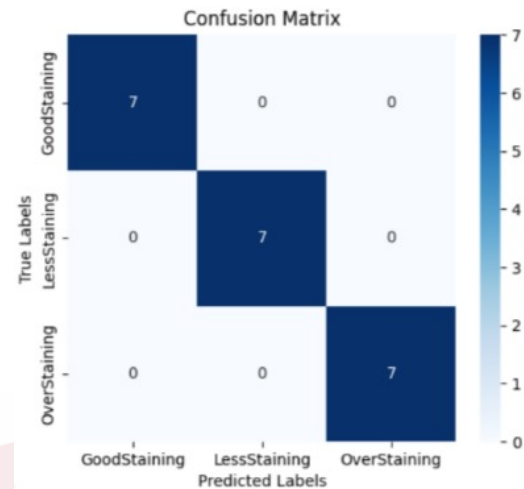
C. Implementasi Model YOLO untuk Deteksi dan Perhitungan Bakteri TBC

Proses pembuatan model deteksi bakteri dilakukan secara bertahap, meliputi pengumpulan data, preprocessing, augmentation, pelatihan, dan evaluasi. Tujuannya adalah mengembangkan sistem deteksi objek yang andal untuk membedakan citra yang mengandung bakteri dan yang tidak mengandung bakteri. Pelatihan model dilakukan menggunakan enam pendekatan seperti yang ditunjukkan pada tabel IX

Tabel IX: Teknik yang Digunakan oleh Model YOLO

Model	Teknik yang Digunakan
1	Pengaturan <i>hyperparameter</i> manual dengan 50 epoch
2	Pengaturan <i>hyperparameter</i> manual dengan 100 epoch
3	<i>Fine tuning</i> Model 2 dengan pengaturan <i>hyperparameter</i> manual dan 100 epoch
4	<i>Fine tuning</i> Model 3 dengan pengaturan <i>hyperparameter</i> manual dan 100 epoch
5	<i>Random search</i> dengan 50 epoch untuk setiap iterasi
6	Menggunakan <i>hyperparameter</i> terbaik dari Model 5 dengan 100 epoch dan <i>early stopping</i>

Pada tabel IX, model 1 menggunakan pengaturan *hyperparameter* secara manual dengan 50 epoch. Model 2 juga menggunakan pengaturan *hyperparameter* manual, namun dengan 100 epoch. Model 3 merupakan hasil *fine tuning* dari Model 2 dengan pengaturan *hyperparameter* manual dan 100 epoch, sedangkan Model 4 merupakan hasil *fine tuning* dari Model 3 dengan pengaturan yang sama. Model 5 menggunakan metode *random search* dengan 50 epoch pada setiap iterasi. Terakhir, Model 6 menggunakan *hyperparameter* terbaik dari Model 5 dengan 100 epoch dan menerapkan *early stopping*.



Gambar 6: Confusion Matrix menggunakan Data Uji

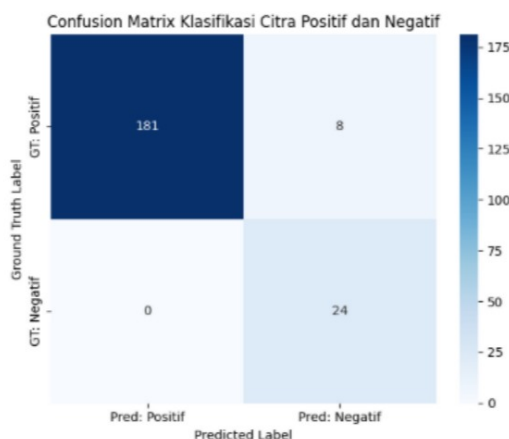
Dataset yang digunakan terdiri dari 576 citra, dengan 238 sampel positif dan 238 sampel negatif sesuai standar *International Union Against Tuberculosis and Lung Disease* (IUATLD). Data ini menjadi dasar dalam pelatihan model. Pembagian data dilakukan dengan proporsi 70% untuk pelatihan, 20% untuk validasi, dan 10% untuk pengujian, guna memastikan pelatihan model yang optimal serta evaluasi pada data yang belum pernah dilihat sebelumnya. Untuk memperkaya data pelatihan, dilakukan proses *augmentation* sehingga jumlah data pelatihan meningkat menjadi 974 citra. Proses *augmentation* ini dilakukan menggunakan Roboflow, yang juga mempermudah proses ekspor dataset.

D. Hasil Pengujian

1) *Pengujian Model CNN* : Setelah proses optimasi tahap kedua, model kemudian dievaluasi untuk mengukur kinerjanya. Evaluasi dilakukan dengan menggunakan data pengujian (test data) dengan proporsi 10% dari dataset yang telah disiapkan sebelumnya, terpisah dari data pelatihan dan validasi. Untuk mengukur performa, digunakan confusion matrix yang membandingkan hasil prediksi model dengan label sebenarnya. Dari confusion matrix ini, dihitung berbagai metrik evaluasi meliputi akurasi, presisi, sensitivitas (recall), spesifisitas, dan F1-skor. Langkah ini memastikan bahwa model tidak hanya memiliki akurasi tinggi, tetapi juga mampu memberikan prediksi yang seimbang antara deteksi positif dan negatif. Setelah evaluasi klasifikasi selesai, tahap berikutnya adalah pengujian skenario deteksi dan perhitungan jumlah bakteri TBC secara otomatis, guna memastikan integrasi model dapat berjalan optimal pada aplikasi yang dikembangkan. Gambar 6 terlihat confusion matrix dari pengujian model. Setiap baris menunjukkan label aktual, sedangkan setiap kolom menunjukkan prediksi model. Terlihat bahwa seluruh sampel dari setiap kelas terklasifikasi dengan benar, dengan nilai diagonal sempurna (7 untuk setiap kelas) dan tanpa prediksi yang salah pada seluruh data uji. Hal ini menunjukkan bahwa model mencapai

Class: LessStaining Precision: 1.0000 Recall (Sensitivity): 1.0000 Specificity: 1.0000 F1-Score: 1.0000	Class: OverStaining Precision: 1.0000 Recall (Sensitivity): 1.0000 Specificity: 1.0000 F1-Score: 1.0000	Class: GoodStaining Precision: 1.0000 Recall (Sensitivity): 1.0000 Specificity: 1.0000 F1-Score: 1.0000
---	---	---

Gambar 7: Hasil Metrik Evaluasi



Gambar 8: Confusion matrix membedakan citra negatif dan positif pada Pengujian Menggunakan Data Uji

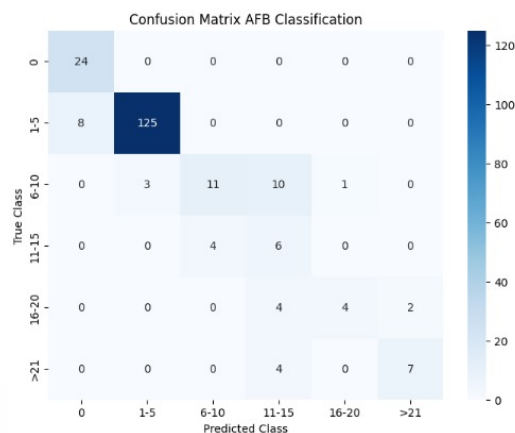
akurasi klasifikasi 100% pada data uji untuk semua kelas: GoodStaining, LessStaining, dan OverStaining.

Gambar 7 ditampilkan metrik evaluasi untuk setiap kelas yang meliputi precision, recall (sensitivity), specificity, dan F1-skor. Untuk ketiga kelas *LessStaining*, *OverStaining*, dan *GoodStaining* seluruh metrik memperoleh nilai sempurna 1.0 atau 100%. Hal ini menunjukkan bahwa model tidak hanya berhasil mengidentifikasi semua kelas dengan benar, tetapi juga melakukannya dengan presisi dan sensitivitas tinggi, tanpa false positive maupun false negative.

2) *Pengujian Model YOLO*: Semua model diuji berdasarkan performa loss, recall, akurasi, F1-skor, mAP50, serta perbandingan kurva loss pada data train dan valid selama pelatihan. Model dengan performa terbaik kemudian diuji berdasarkan kemampuannya dalam mengklasifikasikan citra positif dan negatif, dengan hasil pengujian ditampilkan dalam bentuk confusion matrix. Pengujian selanjutnya dilakukan untuk menilai kemampuan model dalam mendeteksi dan menghitung jumlah AFB dalam satu citra.

Berdasarkan tabel X, kami memilih model pertama sebagai model yang memiliki performa terbaik, lalu kami mengevaluasi model deteksi menggunakan data testing dengan proporsi 10% dari dataset yang terbagi menjadi 6 kategori yaitu citra dengan 0, 1-5, 6-10, 11-15, 16-20, dan > 21 AFB dengan masing masing kategori berjumlah 24, 133, 25, 10, 10, 11 citra. Berikut merupakan Confusion Matrix pada model yang sudah diuji berdasarkan kemampuan membedakan citra positif dan negatif.

Confusion matrik yang ada di Gambar 8 memperjelas seberapa baik model mampu mengklasifikasikan data secara akurat serta tidak ada bias terhadap kelas tertentu. Pada hasil analisis gambar diatas, model mampu memprediksi dengan benar sebanyak 181 citra positif dan 24 citra negatif. Hasil



Gambar 9: Confusion Matrix dari Perhitungan Bakteri menggunakan Data Uji

tersebut menunjukkan bahwa model memiliki kemampuan yang sempurna untuk mengklasifikasikan data yang diberikan.

Dari Tabel perhitungan di atas, diketahui bahwa model memiliki akurasi 96,2%, presisi 100%, sensitivitas 95.7%, spesifisitas 100%, dan F1 skor 97.8%. Model berhasil mendeteksi citra positif sebanyak 181 gambar, citra negatif sebanyak 24 gambar, dan salah mendeteksi 8 gambar yang harusnya citra positif namun dideteksi sebagai citra negatif. Tahap terakhir adalah menilai kemampuan model dalam menghitung jumlah bakteri, dengan hasil yang dikelompokkan ke dalam rentang tertentu. Evaluasi ini bertujuan untuk melihat seberapa dekat prediksi model dengan jumlah bakteri yang sebenarnya. Hasil menunjukkan bahwa akurasi model mencapai lebih dari 80%.

Berdasarkan hasil pada Tabel dan Gambar, model menunjukkan kinerja yang sangat andal dalam mendeteksi citra dengan jumlah *acid-fast bacilli* (AFB) sedikit serta citra negatif. Namun, performa mulai menurun pada kelas 6–10 dan 11–15, di mana banyak AFB yang tidak terdeteksi atau salah diklasifikasikan. Pada kelas 16–20 dan >21, model memperlihatkan kinerja sedang dan cenderung jarang mengklasifikasikan citra sebagai memiliki AFB dalam jumlah besar. Meskipun demikian, model masih cukup sering gagal mengenali citra yang benar-benar mengandung AFB dalam jumlah besar, sehingga sensitivitas pada kelas ini relatif rendah.

3) *Pengujian User Acceptance Testing (UAT)* : Pengujian pada aplikasi dilakukan melalui User Acceptance Testing (UAT) dengan jenis Open Beta Testing. UAT merupakan pengujian yang dilakukan oleh end user yang berinteraksi langsung dengan sistem. Dalam pengujian ini, ditentukan sebanyak 30 responden untuk mencoba aplikasi pada perangkat masing-masing. Pengisian survei menggunakan skala Likert dengan 5 kategori, yaitu 1 menunjukkan sangat tidak nyaman dan 5 menunjukkan sangat nyaman, sebagaimana ditunjukkan pada tabel. Nilai skor ideal dihitung menggunakan formula:

Tabel X: Informasi dari Pelatihan Terakhir dari Masing-masing Model

Model	Nama	Precision	Recall	mAP50	mAP50-95	Epoch
1	Best.pt	0.67137	0.55492	0.59639	0.23822	50/50
2	Best3.pt	0.62248	0.58086	0.59501	0.23083	100/100
3	Best4.pt	0.67915	0.55114	0.59181	0.22468	100/100
4	Best5.pt	0.62895	0.58282	0.63720	0.23024	100/100
5	Best6.pt	0.54006	0.51817	0.49432	0.17496	50/50
6	Best7.pt	0.61482	0.53206	0.53497	0.18410	66/100

Tabel XI: Perhitungan Setiap Metrik Klasifikasi Citra Positif dan Negatif

Model	Akurasi	Presisi	Sensitivitas	Spesifisitas	F1-skor	TP	FP	FN	TN
Pertama	0.92	1.00	0.95	1.00	0.97	181	8	0	24

Tabel XII: Perhitungan Mertrik Deteksi Bakteri TBC

Label	Akurasi	Presisi	Sensitivitas	Spesifisitas	F1-Skor	TP	FP	FN	TN
0	0.962	0.750	1.000	0.957	0.857	24	8	0	181
1-5	0.948	0.976	0.940	0.963	0.958	125	3	8	77
6-10	0.897	0.579	0.440	0.957	0.500	11	8	14	170
11-15	0.897	0.250	0.600	0.911	0.353	6	18	4	180
16-20	0.967	0.800	0.400	0.995	0.533	4	1	6	202
>21	0.972	0.778	0.636	0.990	0.700	7	2	4	200

Skor Ideal = (Jumlah Kategori Skala Likert) × (Jumlah Responden) (6)

Untuk menghitung nilai persentase (P) yang didapat pada setiap jawaban responden adalah dengan formula berikut:

$$P = \frac{\text{Total Nilai}}{\text{Skor Ideal}} \times 100\% \quad (7)$$

Setelah seluruh data dianalisis, aplikasi TBClassify memperoleh skor sebesar 135 dari 150 atau 90%, yang termasuk dalam kategori penerimaan tinggi, menunjukkan tingkat kenyamanan dan usability yang baik serta dapat diterima oleh pengguna.

V. KESIMPULAN

Hasil evaluasi menunjukkan bahwa klasifikasi kualitas pewarnaan ZN menggunakan model CNN mampu memberikan performa sempurna pada seluruh label (Good Staining, Less Staining, dan Over Staining), dengan recall, specificity, dan F1-skor masing-masing mencapai nilai maksimum 1.0 tanpa adanya kesalahan prediksi. Keunggulan ini dipengaruhi oleh penerapan regulasi yang efektif sehingga model tetap stabil terhadap data baru, meskipun akurasi pelatihan tidak sepenuhnya sempurna. Meski demikian, diperlukan validasi lebih lanjut dengan data eksternal untuk memastikan hasil sempurna ini bukan akibat kemiripan distribusi data atau keterbatasan sampel uji. Sementara itu, deteksi bakteri TBC menggunakan model YOLO menunjukkan akurasi tinggi dalam klasifikasi citra positif dan negatif, dengan akurasi 96,2%, precision 100%, recall 95,7%, dan specificity 100%. Namun, kinerjanya menurun signifikan saat mengelompokkan citra berdasarkan jumlah bakteri, khususnya pada kategori menengah (11-15 dan 16-20), yang mengindikasikan kesulitan dalam mendeteksi objek kecil yang saling tumpang tindih

atau padat. Perbaikan dapat dilakukan melalui augmentasi spasial yang lebih variatif, peningkatan resolusi citra atau adaptasi arsitektur model agar lebih sensitif terhadap objek berukuran kecil. Dari sisi pengujian keseluruhan fitur, aplikasi TBClassify menunjukkan kinerja baik dengan nilai User Acceptance Test (UAT) sebesar 90,3%, menandakan mayoritas pengguna puas terhadap performa dan fungsionalitasnya. Fitur utama seperti registrasi, login, dan klasifikasi dapat berjalan lancar pada OS Windows, namun belum mendukung MacOS dan Ubuntu. Dari segi performansi, waktu instalasi dan analisis, dimana RAM sangat memengaruhi waktu respon sistem.

PUSTAKA

- [1] Kementerian Kesehatan Republik Indonesia, "Data Kondisi TBC," <https://www.tbindonesia.or.id/pustaka/tbc/data-kondisi-tbc/>, 2025, diakses pada 4 Agustus 2025.
- [2] World Health Organization, *Global Tuberculosis Report 2023*. Geneva: World Health Organization, 2023.
- [3] Kementerian Kesehatan Republik Indonesia, *Petunjuk Teknis Penatalaksanaan Tuberculosis Resistan Obat di Indonesia*. Jakarta: Kementerian Kesehatan RI, 2020.
- [4] J. G. Jang and J. H. Chung, "Diagnosis and treatment of multidrug-resistant tuberculosis," *Yeungnam University Journal of Medicine*, vol. 37, no. 4, pp. 277–285, Oct. 2020.
- [5] Kementerian Kesehatan Republik Indonesia, "Strategi nasional penanggulangan tuberkulosis di indonesia 2020–2024," <https://yankes.kemkes.go.id/>, 2020, diakses pada 4 Agustus 2025.
- [6] World Health Organization, *WHO Consolidated Guidelines on Tuberculosis. Module 2: Screening – Systematic Screening for Tuberculosis Disease*. Geneva: World Health Organization, 2021.
- [7] A. L. Garcia-Basteiro *et al.*, "Point of care diagnostics for tuberculosis," *Pulmonology*, Jan. 2018.
- [8] K. R. Steingart *et al.*, "Fluorescence versus conventional sputum smear microscopy for tuberculosis: A systematic review," *The Lancet Infectious Diseases*, 2006.
- [9] T. H nscheid, "The future looks bright: Low-cost fluorescent microscopes for detection of *Mycobacterium tuberculosis* and coccidia," *Transactions of the Royal Society of Tropical Medicine and Hygiene*, vol. 102, no. 6, pp. 520–521, Jun. 2008.

- [10] R. Yudiana, Z. Zulmansyah, and H. Garna, "Hubungan kepatuhan terapi obat anti tuberkulosis kombinasi dosis tetap (oat-kdt) dengan kesembuhan pasien tuberkulosis paru dewasa di puskesmas patokebe-usi subang," *Jurnal Integrasi Kesehatan & Sains*, vol. 4, no. 1, pp. 44–49, Jan. 2022.
- [11] S. U. Basundari, S. Harun, R. Ekowatiningsih, and E. Yuwarni, "Uji validitas teknik pcr (polymerase chain reaction) dan pemeriksaan mikroskopis bakteri tahan asam sebagai alat diagnosis penderita tb paru di rumah sakit persahabatan, jakarta," *Media Penelitian dan Pengembangan Kesehatan*, vol. 12, no. 3, 2002.
- [12] A. Tjokronegoro and A. Yusuf, *Tuberkulosis paru: pedoman penataan diagnostik dan terapi*. Fakultas Kedokteran Universitas Indonesia, Jakarta, 1985.
- [13] M. Misnarliah and M. Mudrika, "Pengaruh penundaan pewarnaan preparat bakteri tahan asam metode ziehl neelsen terhadap hasil pemeriksaan mikroskopis," *Jurnal Teknosains Kodepena*, vol. 1, no. 2, pp. 58–63, 2021.
- [14] H. R. da Silva and Y. Maulani, "Perbandingan hasil pemeriksaan mikroskopis sputum bta terhadap metode pcr (genexpert) pada pasien tuberkulosis paru," *Plenary Health: Jurnal Kesehatan Paripurna*, vol. 1, no. 3, pp. 455–460, 2024.
- [15] A. Achmadi, M. M. Mardiah, and S. Wahyu, "Penerapan pemantapan mutu internal terhadap kualitas sediaan pewarnaan ziehl nielsen untuk deteksi mycobacterium tb," *Jurnal Ilmiah Kesehatan (JIKA)*, vol. 3, no. 3, pp. 124–133, 2021.
- [16] IBM Team. (2025) Apa itu convolutional neural network? [Online]. Available: <https://www.ibm.com/id-id/think/topics/convolutional-neural-networks>
- [17] IONOS Editorial Team. (2025) Convolutional neural networks. IONOS. [Online]. Available: <https://www.ionos.com/digitalguide/websites/web-development/convolutional-neural-networks/>
- [18] GeeksforGeeks Editorial Team. (2025) Understanding the confusion matrix in machine learning. GeeksforGeeks. [Online]. Available: <https://www.geeksforgeeks.org/machine-learning/confusion-matrix-machine-learning/>
- [19] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis and machine vision*. Springer, 2013.
- [20] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 11, pp. 3212–3232, 2019.
- [21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
- [22] J. Du, "Understanding of object detection based on cnn family and yolo," in *Journal of Physics: Conference Series*, vol. 1004. IOP Publishing, 2018, p. 012029.
- [23] R. Khanam and M. Hussain, "Yolov11: An overview of the key architectural enhancements. arxiv 2024," *arXiv preprint arXiv:2410.17725*, 2024.
- [24] R. Khanam, M. Hussain, R. Hill, and P. Allen, "A comprehensive review of convolutional neural networks for defect detection in industrial applications," *IEEE Access*, 2024.
- [25] Annisadev. (2024, November) Mengenal roboflow: Solusi lengkap untuk pengembangan model computer vision. Annisadev. [Online]. Available: <https://www.annisadev.com/news/read/1012/mengenal-roboflow-solusi-lengkap-untuk-pengembangan-model-computer-vision.html>
- [26] DQLab. (2024) Hyperparameter tuning dalam proses pemodelan data. DQLab. [Online]. Available: <https://www.dqlab.id/>
- [27] D. Kurniadi, R. M. Shidiq, and A. Mulyani, "Perbandingan penggunaan optimizer dalam klasifikasi sel darah putih menggunakan convolutional neural network," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 14, no. 1, pp. 77–86, 2025.
- [28] P. S. Foundation. (2024) Apa itu python? Python.org. [Online]. Available: <https://www.python.org/doc/essays/blurb/>
- [29] J. Salendah, P. Kalele, A. Tulenan, and J. S. R. Joshua, "Penentuan beasiswa dengan metode fuzzy tsukamoto berbasis web scholarship determination using web based fuzzy tsukamoto method," in *Seminar Nasional Ilmu Komputer (SNASIKOM)*, vol. 2, no. 1, 2022, pp. 81–90.
- [30] R. Ramadhan, E. Fitria, and R. Rosdiana, "Deteksi mycobacterium tuberculosis dengan pemeriksaan mikroskopis dan teknik pcr pada penderita tuberkulosis paru di puskesmas darul imarah," *Sel Jurnal Penelitian Kesehatan*, vol. 4, no. 2, pp. 73–80, 2017.
- [31] S. Aulia, A. B. Suksmono, T. R. Mengko, and B. Alisjahbana, "A novel digitized microscopic images of zn-stained sputum smear and its classification based on iuatld grades," *IEEE Access*, vol. 12, pp. 51 364–51 380, 2024.
- [32] Trivusi. (2022, August) Apa bedanya epoch dan batch size pada deep learning? Trivusi. [Online]. Available: <https://www.trivusi.web.id/2022/08/epoch-dan-batch-size.html>
- [33] A. Sejati and L. Sofiana. (2015) Jurnal kesehatan masyarakat. [Online]. Available: <http://journal.unnes.ac.id/nju/index.php/kemas>