

Deployment of Kidney Tumor Disease Object Detection Using CT-Scan with YOLOv5

Hasyantoko Dwiki Kahingide^{1*}, Abu Salam^{2**}

* Teknik Informatika, Universitas Dian Nuswantoro
111202012979@mhs.dinus.ac.id¹, abu.salam@dsn.dinus.ac.id²

Article Info

Article history:

Received 2024-06-11

Revised 2024-06-15

Accepted 2024-06-19

Keyword:

Kidney tumor detection,
CT-Scan images,
Tumor Ginjal,
YOLOv5,
Object detection,
Intergrasi Web Flask.

ABSTRACT

Image processing plays a crucial role in identifying kidney tumors through CT-Scan images. Object detection technology, particularly YOLO, stands out for its speed and accuracy in facilitating more detailed analysis. Using Flask as a web framework offers optimal responsiveness, providing adaptive ease of use, especially in medical image processing. Evaluation of the model shows impressive results, with a mean Average Precision (mAP) of 0.987 for the 'kidney tumor' label. Detection on public data demonstrated high performance with accuracy, precision, recall, and F1-Score of 98.56%, 98.66%, 99.66%, and 99.16%, respectively. This study also utilized clinical data comprising 62 CT-Scan images. Evaluation of the clinical data revealed that YOLOv5 produced an accurate detection model with accuracy, precision, recall, and F1-Score of 95.16%, 96.72%, 98.33%, and 97.52%, respectively. The research shows that both public and clinical data models can accurately detect kidney tumors based on CT-Scan images. The deployment process using the Flask web-based platform allows direct interaction with users through an intuitive interface, enabling users to upload their CT-Scan images and quickly obtain detection results. These test results provide evidence that object detection using YOLOv5 achieves high accuracy in detecting both public and clinical datasets.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Saat ini, pengolahan citra memiliki suatu aplikasi yang luas dalam berbagai bidang, seperti arkeologi, astronomi, dan bidang medis termasuk penggunaan citra *CT scan* dalam biomedis. Deteksi objek dalam pengolahan citra memiliki peran penting karena mengidentifikasi dan melokalisasi objek-objek penting dalam citra tersebut.[1] Tujuannya adalah mempermudah analisis bagi pengamat citra dengan menyoroti objek-objek yang relevan dalam citra tersebut. Deteksi objek digunakan dalam berbagai aplikasi, terutama dalam bidang medis seperti pengolahan citra *CT scan* untuk memeriksa ginjal manusia [1].

Ginjal adalah organ penting dalam sistem kemih manusia[2]. Manusia memiliki dua buah ginjal yang berada di bagian belakang perut antara tulang punggung bagian bawah dan tengah. Fungsi utama ginjal adalah mengeluarkan kelebihan air, garam, dan zat-zat yang tidak dibutuhkan dari

tubuh melalui pembuangan urin. Salah satu masalah yang bisa terjadi pada ginjal adalah tumor [3].

Tumor ginjal adalah jenis penyakit kanker yang sering ditemukan. Penyakit ini bermula dari sel-sel pada tubulus renal yang bertanggung jawab untuk menyaring limbah dalam darah [3]. Data statistik menunjukkan bahwa kanker ginjal lebih sering terjadi pada pria daripada wanita, dengan dua kali lipat lebih banyak kasus terjadi pada pria. Lebih dari setengahnya, sekitar 59%, kasus kanker ginjal terjadi di negara-negara berkembang[4]. Tingkat insiden kanker ginjal tertinggi tercatat di Amerika Utara dan Eropa, sementara tingkat terendah terlihat di wilayah Asia dan Afrika.

Salah satu alat pemeriksaan yang dapat digunakan untuk mengetahui penyakit tumor ginjal adalah *CT-Scan*. *CT-Scan* (*Computed Tomography*)[5] pada dasarnya hampir mirip dengan perangkat radiografi konvensional. Keduanya saling memanfaatkan kedalaman sinar radiasi untuk melewati objek dan membuat gambar. Yang membedakan dari keduanya adalah cara mereka menghasilkan citra dan tampilan citranya.

Dibandingkan dengan teknik radiografi, citra yang dihasilkan oleh *CT-Scan* tidak tumpang tindih, sehingga lebih mudah untuk dianalisis[6].

Dalam mengidentifikasi area atau lokasi tumor ginjal hanya dengan melihat hasil *CT scan* secara langsung dan berdasarkan tampilan visual ini berpotensi menghasilkan diagnosis yang kurang akurat[7], terutama dalam kasus yang mempengaruhi kelangsungan hidup pasien dengan tumor ginjal. Oleh karena itu, sangat penting memiliki metode yang dapat mengevaluasi, mengidentifikasi, dan mendeteksi area yang terkena tumor dari data *CT scan* pasien dengan penyakit tumor ginjal[8].

Pada saat ini, Era pengenalan dan deteksi objek dalam *computer vision* sedang mengalami perkembangan dengan pesat hingga mulai diterapkan pada berbagai bidang, termasuk bidang industri dan medis[1]. Dengan terus berkembangnya *Deep learning*, bermunculan model-model baru dalam pengenalan dan deteksi objek pada *Computer Vision*. Ini termasuk perkembangan dari model-model seperti *Region-based Convolutional Neural Networks*(R-CNN)[9], *You only look once* (YOLO) hingga *RetinaNet*[10]. Seiring dengan itu, teknik deteksi objek juga menjadi fokus penting dalam pengembangan ini.[11]

You Only Look Once atau yang biasa disebut YOLO merupakan sebuah algoritma dalam *computer vision* yang lebih cepat dalam mengenali objek daripada metode seperti R-CNN dan DPM[12]. Dengan YOLO, hasil deteksinya bisa mencapai 45 frame per detik (*fps*).[10]Ini berarti YOLO mampu untuk mengidentifikasi objek-objek dalam gambar dengan kecepatan yang tinggi, mengungguli kecepatan metode lain seperti R-CNN dan DPM.

Hingga saat ini YOLO memiliki banyak versi dari pengembangan YOLO v1 hingga YOLO v8[11], Tetapi pada penelitian kali ini penulis menggunakan YOLO v5. Pada penelitian sebelumnya[13] juga dilakukan dengan menggunakan YOLO v5 [14]mendapatkan kesimpulan bahwa model yang diusulkan secara efektif mengelompokkan tumor ginjal dalam beberapa gambar *CT-Scan*, distribusi tidak teratur dan perbedaan dari variabel bentuk tumor ginjal[15].

Deteksi objek dengan *Deep learning* adalah proses komputer yang memungkinkan sistem untuk mengidentifikasi dan mengklasifikasikan objek dalam gambar atau video[16]. *Deep learning*, sebagai bagian dari kecerdasan buatan dan *machine learning*[17], menggunakan jaringan neural yang mirip otak manusia untuk memproses data dan mempelajari pola[18]. Dalam deteksi objek, *Deep learning* menggunakan algoritma seperti CNN yang efektif dalam mengidentifikasi fitur-fitur objek seperti bentuk, warna, dan ukuran[19].

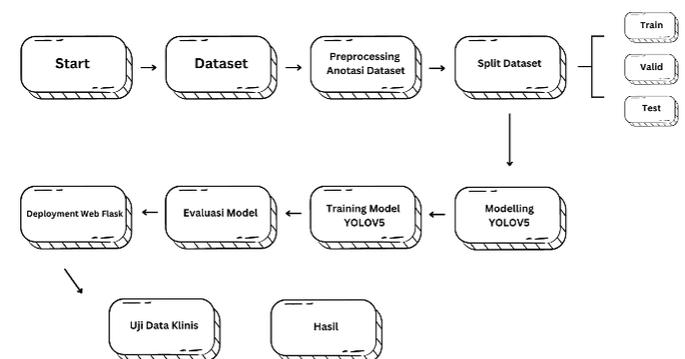
Dalam penelitian ini, peneliti menggunakan *Flask* yang berbasis web untuk *deployment*. *Flask* adalah *framework* berbasis *Python*[20] yang tidak memerlukan library tambahan. Dibandingkan dengan *Django*, *Flask* dianggap lebih "*Python ic*"[21] karena aplikasi web yang dibuat dengan *Flask* cenderung lebih dijelaskan secara gamblang dalam penggunaannya[22]. Hal ini membuatnya menjadi pilihan

yang lebih mudah untuk pemula karena membutuhkan respon yang lebih *responsive*.

Pengolahan citra memiliki peran krusial dalam identifikasi dan analisis tumor ginjal menggunakan citra *CT-Scan*. [17] Deteksi objek menjadi aspek penting dalam mengidentifikasi area yang terkena tumor, memungkinkan analisis yang lebih spesifik[1]. Dalam pengembangan ini, teknik *Deep learning* seperti YOLO v5[13] menonjol karena kecepatan dan ketepatannya dalam mendeteksi tumor pada citra medis. YOLO v5 [23]mampu mengidentifikasi tumor ginjal dengan akurasi tinggi dalam gambar *CT-Scan*, mempercepat proses diagnosis. Selain itu, penelitian ini juga menguji data klinis (privat), yang memungkinkan validasi model pada data nyata yang lebih relevan dengan kasus-kasus klinis. Penerapan *Flask* sebagai *framework* web untuk *deployment*[24] real-time menunjukkan responsivitas yang tinggi,[25] memfasilitasi penggunaan yang lebih mudah, dan sesuai untuk proyek-proyek pengolahan citra medis dari level pemula hingga tingkat lanjutan. Kombinasi teknologi ini memberikan peluang yang luas dalam penelitian dan pengembangan di bidang pengolahan citra medis.

II. METODE PENELITIAN

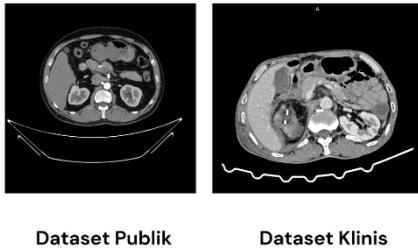
Diagram alur untuk metodologi dalam penelitian ini dapat dilihat di pada gambar berikut.



Gambar 1. Metode Penelitian

A. Dataset

Dalam penelitian ini, Peneliti bekerja sama dengan radiografer atau institusi medis untuk mendapatkan akses ke dataset *CT scan* ginjal. Dalam penelitian ini, melibatkan peran radiografer Rumah Sakit Bhayangkara AKPOL Semarang untuk proses anotasi dan validasi Radiografer Isa Faiz Muammar (NIR 3374231222097). Dataset klinis/privat berasal dari RSUD Kebumen dan RS Sultan Agung Semarang. Menggunakan sample sebanyak 10% dari total 628 citra, berformat *JPEG* dan hanya data yang terdiagnosis tumor ginjal untuk data testing. Untuk data training, dalam penelitian ini menggunakan dataset publik dari platform Kaggle, *Kidney CT-Scan*. Dengan jumlah keseluruhan 2100 citra dengan format *JPG*. Untuk testing menggunakan 209 citra.



Gambar 2. Dataset Tumor Ginjal

Gambar 2 menunjukkan contoh citra pada Dataset Publik dan Dataset Klinis. Pada tahap awal penelitian, dilakukan evaluasi kualitas data, khususnya untuk citra *CT scan* yang akan digunakan. Proses ini melibatkan pemeriksaan yang teliti terhadap resolusi dan kejelasan setiap citra. Resolusi yang tinggi diperlukan agar semua detail halus terlihat dengan jelas, terutama dalam hal ukuran, bentuk, dan lokasi tumor ginjal

B. Preprocessing Anotasi Dataset

Langkah selanjutnya adalah melakukan anotasi dataset secara manual menggunakan Roboflow, Objek yang dianotasi adalah tumor ginjal pada citra *CT-Scan*.



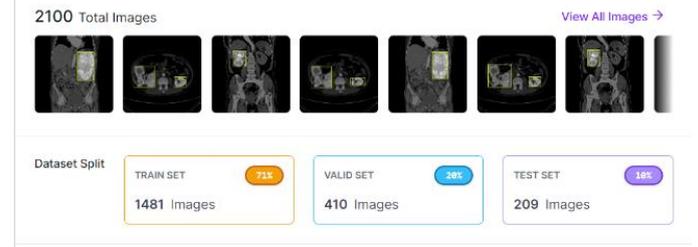
Gambar 3. Anotasi Manual Bounding box

Proses ini melibatkan identifikasi dan penandaan setiap tumor ginjal untuk memberikan label yang konsisten pada semua citra dalam dataset. Anotasi yang akurat memungkinkan model *machine learning* dilatih untuk mendeteksi dan mengenali tumor ginjal dalam citra *CT-Scan* dengan akurasi tinggi.

C. Pembagian Data / Split Dataset

Di Proses Pembagian data / Split Data dataset akan dibagi menggunakan Roboflow menjadi tiga bagian utama: set pelatihan, set pengujian, dan set validasi. yang mempermudah proses pembagian data dengan menyediakan interface yang intuitif. Set pelatihan berperan sebagai fondasi utama dalam proses pelatihan model, di mana citra-citra digunakan untuk mengajari model mengenali dan mendeteksi ciri-ciri tumor ginjal seperti bentuk, ukuran, dan lokasi tumor dalam citra CT scan. Melalui proses iteratif, model terus meningkatkan kemampuannya dalam memahami dan menafsirkan data ini dengan efektif. Set validasi digunakan untuk memantau dan

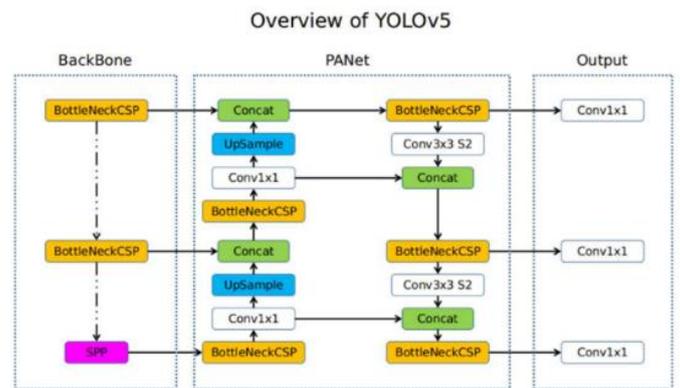
mengevaluasi kinerja model selama pelatihan, dan memastikan generalisasi model. Set pengujian digunakan sebagai alat ukur akhir untuk menilai kinerja model secara keseluruhan, menguji akurasi dan efektivitas model pada data baru yang tidak pernah digunakan dalam proses pelatihan atau validasi sebelumnya. Dengan pembagian data menjadi set pelatihan, validasi, dan pengujian menggunakan Roboflow, kami memastikan bahwa model deteksi tumor ginjal yang dikembangkan memiliki kemampuan generalisasi yang baik dan dapat diandalkan untuk aplikasi dunia nyata.



Gambar 4. Split Dataset

D. Model

Arsitektur *YOLOV5* mencakup tiga elemen utama: *Backbone*, *Neck*, dan *Head*. *Backbone*, yang terdiri dari *BottleNeckCSP* dan *SPP*, berfungsi untuk ekstraksi fitur dengan tingkat efisiensi dan presisi yang tinggi. *Neck* menggunakan *Concat*, *UpSample*, *Conv1x1*, dan *Conv3x3* untuk menggabungkan dan memproses fitur-fitur tersebut. *Head*, yang menggunakan *Conv1x1*, bertugas mendeteksi objek dengan mengonversi fitur menjadi jumlah kelas yang diperlukan dan menghitung kotak pembatas. Dengan demikian, integrasi yang kuat dari ketiga komponen ini memungkinkan *YOLOV5* menyajikan deteksi objek dengan kecepatan dan akurasi yang tinggi.[13].



Gambar 5. Arsitektur YOLOV5

YOLOV5 memiliki beberapa model yaitu: *YOLOv5n*, *YOLOv5s*, *YOLOv5m*, *YOLOv5l*, *YOLOv5x*. Penggunaan model *YOLOv5n* untuk system aplikasi yang ringan, sedangkan penggunaan web dan mobile direkomendasikan menggunakan *YOLOv5s* atau *YOLOv5m* karena ukurannya dan kecepatannya yang lebih tinggi, dan performa saat model

ini dideploy. Adapun untuk penggunaan cloud direkomendasikan menggunakan YOLOv5l atau YOLOv5x.

Dalam penelitian ini, peneliti menggunakan model YOLOv5m karena model ini sesuai untuk digunakan dalam aplikasi web. YOLOv5m dipilih karena ukurannya yang lebih kecil dan kecepatannya yang lebih tinggi, sehingga memberikan performa optimal saat diimplementasikan pada web deployment.

	Nano	Small	Medium	Large	XLarge
	YOLOv5n	YOLOv5s	YOLOv5m	YOLOv5l	YOLOv5x
	4 MB _{FP16}	14 MB _{FP16}	41 MB _{FP16}	89 MB _{FP16}	166 MB _{FP16}
	6.3 ms _{V100}	6.4 ms _{V100}	8.2 ms _{V100}	10.1 ms _{V100}	12.1 ms _{V100}
	28.4 mAP _{COCO}	37.2 mAP _{COCO}	45.2 mAP _{COCO}	48.8 mAP _{COCO}	50.7 mAP _{COCO}

Gambar 6. Model YOLO

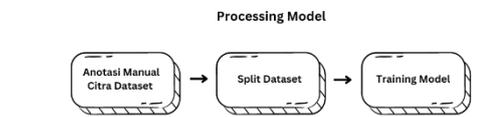
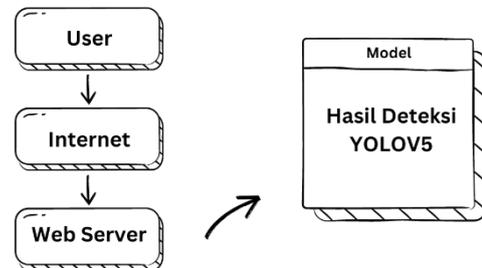
E. Proses Training

Dataset yang sudah diekspor dari Roboflow akan digunakan pada proses perancangan model deteksi. Setelah dataset diekspor, langkah selanjutnya adalah melakukan pelatihan model dengan menggunakan pre-model YOLOv5 yang diunduh dari repositori resmi di *GitHub*, dan dilakukan penginstalan package yang diperlukan. Model di-training dengan memakai Google Colab dan menggunakan GPU Tesla T4 untuk melakukan training. Dataset Roboflow sebelumnya akan di-import dan dilakukan training model dengan memanggil `train.py` dan mengisi parameter yang diperlukan. Parameter pada `train.py` terdiri atas `img` untuk menentukan size image, size image yang digunakan adalah 640. Parameter `batch` digunakan untuk menentukan `batch size`, `batch size` yang digunakan secara normal adalah 16. `Epoch` untuk menentukan banyaknya pelatihan yang dilakukan algoritma model, epoch yang digunakan adalah 100. `Data` merupakan parameter untuk lokasi dataset. Selain itu, untuk mengantisipasi overfitting, proses training juga dilengkapi dengan mekanisme `early stopping`. `Early stopping` digunakan untuk menghentikan pelatihan jika performa model pada test set tidak mengalami peningkatan setelah sejumlah epoch tertentu. Ini dilakukan dengan memonitor metrik performa (seperti `loss` atau akurasi) pada test set, dan jika tidak ada peningkatan selama beberapa `epoch` berturut-turut, proses training akan dihentikan lebih awal.

F. Proses Deployment

Setelah proses pelatihan model selesai, langkah berikutnya adalah melanjutkan dengan proses penyebaran (deployment) menggunakan platform berbasis web, Flask. Implementasi YOLOv5 dalam arsitektur deployment dimulai dengan mengintegrasikan model yang telah dilatih ke dalam perangkat lunak yang telah disiapkan sebelumnya. Tahap berikutnya adalah pengujian menyeluruh untuk memastikan bahwa semua komponen berfungsi dengan baik dan tanpa hambatan. Selain itu, dibuat juga antarmuka pengguna yang intuitif dan ramah pengguna, memungkinkan pengguna untuk mengunggah citra CT scan ginjal yang akan dianalisis oleh

sistem. Seluruh proses ini dijalankan pada server VPS, yang sudah dilengkapi dengan semua kebutuhan library seperti Python, Flask, dan model YOLOv5 yang telah dilatih. Dengan demikian, aplikasi Flask ini dapat diakses dengan mudah melalui alamat domain yang telah ditentukan, memberikan kenyamanan dan kemudahan bagi pengguna.



Gambar 7. Arsitektur Deployment Web Flask



Gambar 8. Hasil Deployment menggunakan Flask

Dengan menggunakan Flask, pada halaman utama ini memungkinkan pengguna untuk mengunggah gambar CT-Scan mereka. Antarmuka ini dirancang sederhana dan intuitif, memudahkan pengguna untuk mengunggah gambar dan mendapatkan hasil deteksi dengan cepat. Pengguna memilih file citra CT-Scan dari perangkat mereka menggunakan tombol "Choose File", kemudian menekan tombol "Upload" untuk memulai proses pengunggahan. Setelah citra diunggah, sistem memproses gambar menggunakan model YOLOv5 yang telah dilatih untuk mendeteksi tumor ginjal, dan hasil analisis kemudian ditampilkan kepada pengguna berupa visualisasi deteksi tumor pada gambar. Sistem ini dijalankan pada server VPS yang telah dikonfigurasi dengan semua library yang diperlukan seperti Python, Flask, dan model YOLOv5, sehingga pengguna dapat mengaksesnya melalui alamat domain terkait dengan VPS tersebut. Dengan antarmuka ini, sistem memungkinkan pengguna, seperti radiografer atau profesional medis, untuk dengan mudah mengunggah dan menganalisis citra CT-Scan ginjal.

Gambar 9. Hasil Deteksi *Flask*

Dalam gambar hasil deteksi, terdapat anotasi berbentuk kotak berwarna merah yang menyoroti area spesifik pada citra, menandakan objek atau fitur tertentu yang telah terdeteksi oleh model YOLOv5, dengan label dan skor IoU. Jika nilai IoU tersebut lebih dari 0,7, maka hasil kinerja model untuk gambar tersebut dianggap akurat. Pada gambar tersebut, nilai IoU yang ditampilkan adalah 0,86, menunjukkan bahwa model telah mendeteksi tumor ginjal dengan tingkat akurasi yang tinggi.[26]

G. Evaluasi Model

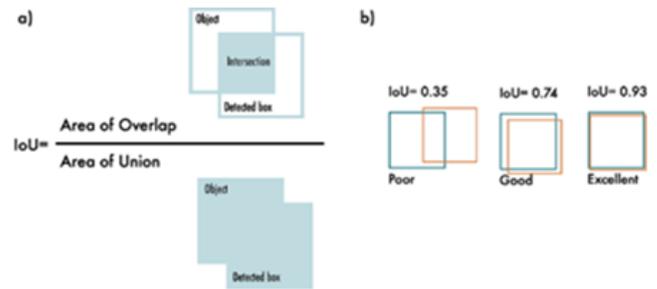
Pada tahap evaluasi model, metode evaluasi yang digunakan adalah IoU (*Intersection over Union*) untuk menilai performa model. IoU adalah ukuran yang memperhitungkan seberapa baik kotak pembatas yang diprediksi oleh model bertumpang tindih dengan kotak pembatas yang sebenarnya. Metrik ini digunakan untuk menghitung *Precision*, yaitu perbandingan antara jumlah objek yang diprediksi dengan benar terhadap total prediksi model. Selain itu, *Recall* juga dihitung, yang merupakan perbandingan antara jumlah objek yang diprediksi dengan benar terhadap total objek yang sebenarnya. Untuk evaluasi yang komprehensif, dilakukan perhitungan *mean average Precision* (mAP), yang menggabungkan nilai *Precision* dan *Recall*. *Precision* (P), *Recall* (R), *average Precision* (AP), dan *mean average Precision* (mAP) adalah metrik penting yang digunakan untuk mengevaluasi kinerja model deteksi objek dalam konteks uji klinis pada dataset klinis.

Intersection over Union adalah nilai berdasarkan statistik kesamaan dan keragaman set sampel bertujuan untuk mengevaluasi area tumpang tindih (area yang beririsan) antara dua *bounding box*, yaitu *bounding box* hasil prediksi model dan *bounding box* data asli. IoU memiliki nilai minimum agar hasil prediksi dianggap benar positif yang disebut IoU threshold. Berikut terdapat persamaan IoU.

$$IoU = \frac{\text{Area of Intersection of two boxes}}{\text{Area of Union of two boxes}}$$

IoU *threshold* merupakan nilai ambang batas nilai IoU suatu citra yang berhasil di deteksi oleh model yang digunakan untuk menentukan apakah *bounding box* termasuk

True positive atau *false positive*. Nilai IoU *threshold* dapat mempengaruhi nilai mAP. IoU *threshold* memiliki nilai antara 0 sampai 1. Jika menggunakan nilai IoU *threshold* yang rendah maka model memiliki jumlah *false positive* yang besar. Sebaliknya, jika menggunakan nilai *threshold* yang lebih tinggi dapat menyebabkan pengurangan *True positive* yang sangat signifikan saat proses training. Pengurangan nilai *threshold* secara tepat dapat meningkatkan kinerja deteksi. Jika pengurangan nilai IoU *threshold* terlalu besar dapat menyebabkan pengurangan yang signifikan pada *positive*.

Gambar 10. Ilustrasi dan Persamaan IoU *Threshold*

Tahap ini bertujuan untuk mengevaluasi kinerja model dengan cara mengukur *Precision*, yaitu rasio antara objek yang diprediksi benar berbanding dengan keseluruhan hasil yang diprediksi oleh model, *Recall*, yaitu rasio antara objek yang diprediksi benar berbanding keseluruhan hasil aktualnya, *mean average Precision* (mAP) dihitung menggunakan nilai *Precision* dan *Recall*. *Precision* (P), *Recall* (R), *average Precision* (AP), *mean average Precision* (mAP).

Ketika menghitung *Precision* (P), *Recall* (R), *average Precision* (AP), dan *mean average Precision* (mAP), langkah awalnya adalah menetapkan definisi untuk *True positive* (TP), *False positive* (FP), *False Negative* (FN), dan *True Negative* (TN). *True positive* (TP) mengindikasikan jumlah objek yang berhasil dideteksi secara akurat oleh model, menunjukkan keberhasilan dalam pengenalan objek yang sebenarnya ada. *False positive* (FP) adalah jumlah objek yang tidak ada namun salah dideteksi oleh model sebagai objek, menggambarkan kesalahan deteksi. *False Negative* (FN) adalah jumlah objek yang sebenarnya ada tetapi tidak terdeteksi oleh model, menunjukkan kegagalan dalam deteksi objek yang ada. *True Negative* (TN) adalah jumlah area yang benar-benar tidak memiliki objek dan tidak salah dideteksi sebagai objek oleh model, menunjukkan ketepatan model dalam mengidentifikasi area tanpa objek.

III. HASIL DAN PEMBAHASAN

A. Evaluasi Dataset Publik

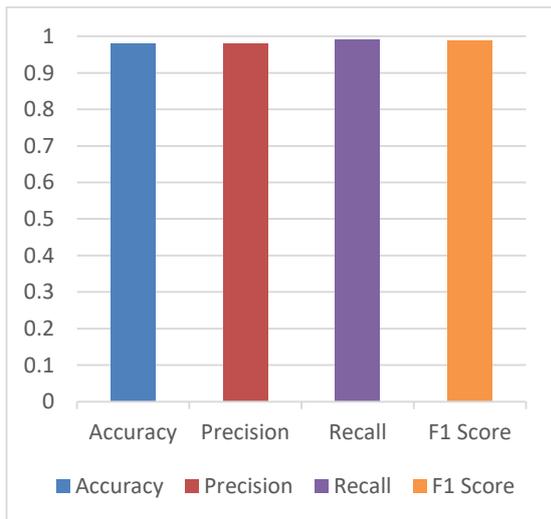
Pengujian ini berfokus pada label tumor ginjal yang dianggap sebagai dataset publik yang dilakukan dengan mencatat data aktual dan membandingkannya dengan data hasil prediksi. Total dataset *Kidney* dari Kaggle yang

digunakan hingga akhir pengujian sebanyak 209 gambar. Untuk pengujian publik dataset, kita menggunakan data testing yang sudah kita split. *Confusion Matrix* sebagai validasi yang didapat dari membandingkan data tersebut dapat dilihat pada Tabel 1 yang merupakan hasil dari keseluruhan deteksi objek dari 209 citra. Citra yang diuji mengidentifikasi adanya tumor ginjal, dengan kemungkinan terdapat lebih dari satu objek terdeteksi pada setiap gambar.

TABEL 1
CONFUSION MATRIX TUMOR GINJAL

Total Keseluruhan			
Jumlah Data : 209		Predicted Values	
		Negatif	Positif
Actual Values	False	1	2
	True	0	206

Pada Tabel 1, didapatkan gambar yang masuk ke database sebanyak TP + FP (206 + 2 = 208). Perhitungan tersebut sesuai dengan pengecekan gambar yang terkirim oleh sistem dari validasi database. Gambar penyakit tumor ginjal yang seharusnya dikirimkan oleh model adalah TP + TN yaitu sebanyak 206 gambar.



Gambar 11. Dataset Publik

Berdasarkan hasil pengujian, model menunjukkan kinerja yang sangat baik dalam mendeteksi objek pada gambar, dengan akurasi sebesar 98.56%. Ini berarti bahwa dari total 209 gambar yang diuji, 98,56% prediksi yang diberikan oleh model adalah benar, baik itu deteksi positif maupun negatif. *Precision* yang diperoleh adalah 98.66%, yang mengindikasikan bahwa dari semua prediksi yang dianggap sebagai objek oleh model, 98.66% di antaranya benar-benar adalah objek. Hal ini menunjukkan bahwa model memiliki tingkat kesalahan yang sangat rendah dalam memberikan prediksi positif palsu (*false positive*). Selain itu, *Recall* model mencapai 99.66%, menandakan bahwa model berhasil mendeteksi 99.66% dari semua kasus yang benar-benar

mengandung objek, dengan hanya sedikit yang terlewat (*false negative*). *F1-Score* yang diperoleh adalah 99.16%, mencerminkan keseimbangan harmonis antara *Precision* dan *recall*, yang menegaskan bahwa model tidak hanya akurat dalam prediksi positif tetapi juga konsisten dalam mendeteksi keberadaan objek secara keseluruhan. Secara keseluruhan, hasil ini menunjukkan bahwa model beroperasi dengan sangat baik, menghasilkan akurasi tinggi dan hanya sedikit kesalahan deteksi

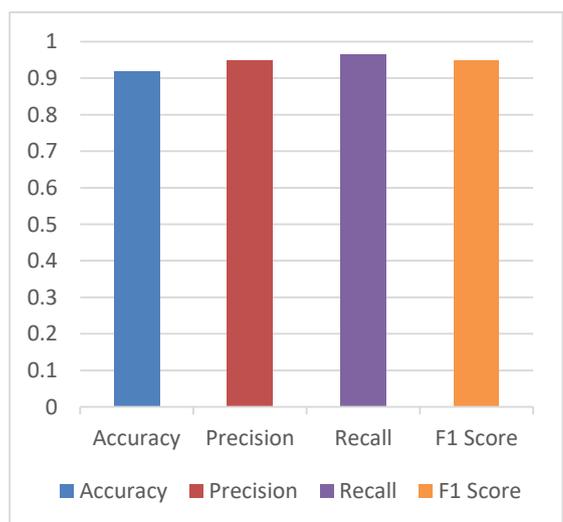
B. Evaluasi Dataset Klinis

Pengujian ini berfokus pada label tumor ginjal yang dianggap sebagai data set privat/klinis yang dilakukan dengan mencatat data aktual dan membandingkannya dengan data hasil prediksi. Total dataset klinis (privat) yang digunakan untuk mengevaluasi model sebanyak 10% dari keseluruhan. *Confusion Matrix* sebagai validasi yang didapat dari membandingkan data tersebut dapat dilihat pada Tabel 2 sebagai berikut.

TABEL 2
CONFUSION MATRIX TUMOR

Total Keseluruhan			
Jumlah Data : 62		Predicted Values	
		Negatif	Positif
Actual Values	False	1	2
	True	0	59

Berdasarkan data yang didapat dari *Confusion Matrix* pada Tabel 1, didapatkan gambar 57 gambar benar-benar tumor ginjal (TP), 3 gambar bukan tumor ginjal (FP), dan 2 gambar tumor ginjal tidak teridentifikasi (FN), serta didapatkan yang masuk ke database sebanyak TP + FP (57 + 0 = 57). Perhitungan tersebut sesuai dengan pengecekan gambar yang terkirim oleh sistem dari validasi database. Gambar penyakit tumor ginjal yang seharusnya dikirimkan oleh model adalah TP + TN yaitu sebanyak 57 gambar.



Gambar 12. Dataset Klinis

Berdasarkan hasil pengujian, model ini menunjukkan kinerja yang baik dalam mendeteksi objek positif dengan akurasi sebesar 95.16%, yang berarti bahwa dari seluruh prediksi yang dibuat oleh model, sekitar 95.16% di antaranya adalah benar. Precision model mencapai 96.72%, menunjukkan bahwa hampir semua deteksi yang dilakukan adalah benar, dengan hanya sekitar 5% prediksi yang salah (False Positives). Model juga memiliki recall sebesar 98.33% yang menunjukkan bahwa sekitar 98.33% dari seluruh objek positif dalam dataset berhasil terdeteksi, sementara sekitar 3.39% terlewat (False Negatives). F1 Score-nya adalah 97.52%, mencerminkan keseimbangan yang kuat antara presisi yang tinggi dan kemampuan mendeteksi objek positif. Secara keseluruhan, hasil ini menunjukkan bahwa model beroperasi dengan kinerja yang sangat baik, terutama dalam mendeteksi objek positif dengan tingkat kesalahan yang rendah.

IV. KESIMPULAN

Berdasarkan hasil pengujian, kedua model menunjukkan kinerja yang sangat baik dalam mendeteksi objek. Model pada data publik mencapai akurasi sebesar 98.56%, dengan Precision dan Recall masing-masing mencapai 98.66% dan 99.66%. F1-Score yang diperoleh adalah 99.16%, mencerminkan keseimbangan harmonis antara Precision dan Recall, menegaskan bahwa model tidak hanya akurat dalam prediksi positif tetapi juga konsisten dalam mendeteksi keberadaan objek secara keseluruhan. Sementara itu, model pada data klinis mencapai akurasi sebesar 95.16%, dengan Precision sebesar 96.72% dan Recall sebesar 98.33%. F1-Score untuk data klinis adalah 97.52%, menunjukkan bahwa model memiliki keseimbangan yang kuat antara presisi yang tinggi dan kemampuan mendeteksi objek positif. Evaluasi menunjukkan bahwa kedua model memiliki kemampuan deteksi yang akurat dengan kesalahan yang minimal.

Model ini berhasil mendeteksi data klinis dan data publik dengan sangat baik. Nilai *mean average Precision* (mAP) yang tinggi pada kedua model menegaskan kemampuan mereka dalam mendeteksi objek dengan tingkat presisi yang tinggi. Integrasi model pada kedua dataset menunjukkan kinerja yang sangat baik, dengan nilai akurasi, Precision, Recall, dan F1-Score yang dihasilkan. Proses deployment menggunakan platform berbasis web Flask memungkinkan interaksi langsung dengan pengguna melalui antarmuka yang intuitif, memungkinkan pengguna untuk mengunggah gambar CT-Scan mereka dan mendapatkan hasil deteksi dengan cepat. Hasil pengujian ini memberikan bukti bahwa deteksi objek menggunakan YOLOV5 berhasil mencapai tingkat akurasi yang tinggi dalam mendeteksi dataset publik maupun klinis.

DAFTAR PUSTAKA

- [1] Q. Aini, N. Lutfiani, H. Kusumah, and M. S. Zahran, "Deteksi Dan Pengenalan Objek Dengan Model *Machine Learning*: Model Yolo," 2021.
- [2] M. Ramanda Hasibuan, "Pemilihan Fitur dengan Information Gain untuk Klasifikasi Penyakit Gagal Ginjal menggunakan Metode Modified K-Nearest Neighbor (MKNN)," 2019. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [3] R. Tri Rahayu, N. Sulaksono, A. Nino Kurniawan, P. Kemenkes Semarang, and I. Corresponding Author, "Studi Kasus Pemeriksaan MSCT Urografi Multiphase dengan Klinis Tumor Ginjal," *Jurnal Imejing Diagnostik (JImeD)*, vol. 9, pp. 75–79, 2023, [Online]. Available: <http://ejournal.poltekkes-smg.ac.id/ojs/index.php/jimed/index>
- [4] F. Aulia *et al.*, "Implementasi Algoritma Convolutional Neural Network Untuk Mendeteksi Penyakit Ginjal Implementation Of Convolutional Neural Network For Detecting *Kidney Disease*," Medan, 2022. [Online]. Available: <http://jtika.if.unram.ac.id/index.php/JTIKA/>
- [5] L. Trisnawati, L. Hakim, and M. Kom, "Segmentasi Citra *Ct Scan* Lung Menggunakan Deteksi Tepi Sobel Dan Metode Distance Regularized Level Set Evolution (DRLSE)," Online, 2018. [Online]. Available: <http://jurnal.yudharta.ac.id/v2/index.php/EXPLORE-IT/>
- [6] O. D. Nurhayati, "Metode Segmentasi Untuk Analisis Citra Digital Head *Ct-Scan*," 2017.
- [7] D. Saputra and Sumijan, "Deteksi Tepi untuk Mengidentifikasi Kanker Ginjal Menggunakan Metode Sobel, Roberts dan Canny," *J Teknol*, pp. 48–54, Dec. 2022, doi: 10.35134/jitekin.v12i2.69.
- [8] X. Liu, L. Song, S. Liu, and Y. Zhang, "A review of deep-learning-based medical image segmentation methods," *Sustainability (Switzerland)*, vol. 13, no. 3, pp. 1–29, Feb. 2021, doi: 10.3390/su13031224.
- [9] Q. Yu, Y. Shi, J. Sun, Y. Gao, J. Zhu, and Y. Dai, "Crossbar-Net: A Novel Convolutional Neural Network for *Kidney Tumor Segmentation* in CT Images," *IEEE Transactions on Image Processing*, vol. 28, no. 8, pp. 4060–4074, Aug. 2019, doi: 10.1109/TIP.2019.2905537.
- [10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016. [Online]. Available: <http://pjreddie.com/YOLO/>
- [11] H. Feng, X. Kou, Z. Tang, and L. Li, "FYU-Net: A Cascading Segmentation Network for *Kidney Tumor Medical Imaging*," *Comput Math Methods Med*, vol. 2022, 2022, doi: 10.1155/2022/4792532.
- [12] M. Anwar, Y. Kristian, and E. Setyati, "Classification Of Chili Plant Diseases Equipped With Leaf And Fruit Image Segmentation Using Yolo V7," *Journal of Information Technology and Computer Science (INTECOMS)*, vol. 6, no. 1, 2023.
- [13] Z. H. Batubara, Y. Hamonangan, M. Arfan, and A. Hidayatno, "Perancangan Sistem Deteksi Pelanggaran Penggunaan Helm Dengan Metode *Deep Learning* Menggunakan *YoloV5* Ultralytic," *Transient: Jurnal Ilmiah Teknik Elektro*, vol. 13, no. 1, pp. 11–20, Mar. 2024, doi: 10.14710/transient.v13i1.11-20.
- [14] N. Benarkah, M. F. Naufal, and B. Renatasiva, "YOLOV5 untuk Menghitung Sel Darah Merah dan Sel Darah Putih," *Keluwih: Jurnal Sains dan Teknologi*, vol. 5, no. 1, pp. 10–18, Apr. 2024, doi: 10.24123/saintek.v5i1.6291.
- [15] "20018-Article Text-40240-1-10-20230508".
- [16] X. Liu, L. Song, S. Liu, and Y. Zhang, "A review of deep-learning-based medical image segmentation methods," *Sustainability (Switzerland)*, vol. 13, no. 3, pp. 1–29, Feb. 2021, doi: 10.3390/su13031224.
- [17] R. Wang, T. Lei, R. Cui, B. Zhang, H. Meng, and A. K. Nandi, "Medical image segmentation using *Deep learning*: A survey," *IET Image Process*, vol. 16, no. 5, pp. 1243–1267, Apr. 2022, doi: 10.1049/ipr2.12419.
- [18] J. Chen and X. Ran, "Deep learning with Edge Computing: A Review," *Proceedings of the IEEE*, vol. 107, no. 8. Institute of Electrical and Electronics Engineers Inc., pp. 1655–1674, Aug. 01, 2019. doi: 10.1109/JPROC.2019.2921977.
- [19] D. Nafis Alfarizi, R. Agung Pangestu, D. Aditya, M. Adi Setiawan, and P. Rosyani, "Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis," 2023. [Online]. Available: <https://jurnalmahasiswa.com/index.php/aidanspk>

- [20] M. Hatami, T. Tukino, F. Nurapriani, W. Widiyawati, and W. Andriani, "Deteksi Helmet Dan Vest Keselamatan Secara Realtime Menggunakan Metode Yolo Berbasis Web Flask," *EDUSAINTEK: Jurnal Pendidikan, Sains dan Teknologi*, vol. 10, no. 1, pp. 221–233, Jan. 2023, doi: 10.47668/edusaintek.v10i1.651.
- [21] H. Shrivastava, "Symptomatic Assistance," *Int J Res Appl Sci Eng Technol*, vol. 9, no. VII, pp. 3658–3660, Jul. 2021, doi: 10.22214/ijraset.2021.37132.
- [22] J. Rathod and K. Trivedi, "Custom Object Detection, Tracking And Web Api With Yolo And Flask Using Darknet Neural Network Framework," *International Research Journal of Engineering and Technology*, 2021, [Online]. Available: www.irjet.net
- [23] S. Gunawan Zain and A. Ardilla, "Detection Of Vehicle Type And License Plate With Convolutional Neural Network Model Yolov7," *Jurnal Teknik Informatika (JUTIF)* 5.2.XX, vol. 5, no. 2, pp. 621–636, 2024, doi: 10.52436/1.jutif.2024.5.2.XX.
- [24] S. Mody, H. Mehta, P. Mantri, B. Ali, and A. Khivasara, "Safety Gear Equipment Detection for Warehouse and Construction Sites Using YOLOv5," *International Research Journal of Engineering and Technology*, 2021, [Online]. Available: www.irjet.net
- [25] "Waste Object Detection Using Yolov5," *International Journal of Progressive Research in Engineering Management and Science*, Apr. 2024, doi: 10.58257/ijprems33165.
- [26] T. Freund, Y. Hamdaoui, and A. Spiwack, "Union and intersection contracts are hard, actually," Jun. 2021, doi: 10.1145/3486602.3486767.