

Pengenalan Karakter Tulisan Tangan pada Dokumen Berita Acara Bimbingan Skripsi menggunakan *Preprocessing* dan *YOLOv8*

Arifah Nur Ainia¹⁾, Nanik Suciati²⁾, Hudan Studiawan³⁾

^{1, 2, 3)} Departemen Teknik Informatika, Institut Teknologi Sepuluh Nopember, Surabaya
E-mail: 6025221061@student.its.ac.id¹⁾, nanik@if.its.ac.id²⁾, hudan@if.its.ac.id³⁾

Abstrak

Pengenalan karakter tulisan tangan merupakan salah satu tantangan dalam bidang pengolahan citra digital, terutama pada dokumen akademik seperti berita acara bimbingan skripsi. Penelitian ini bertujuan untuk membandingkan performa model YOLOv8 dalam mendeteksi karakter tulisan tangan pada dua jenis dataset, yakni gambar original dan gambar hasil preprocessing. Preprocessing dilakukan melalui tahapan grayscale, CLAHE, Gaussian blur, adaptive thresholding Gaussian, dilasi, dan erosi. Label pada data preprocessing diperoleh dengan menyalin anotasi dari data original tanpa melakukan penyesuaian terhadap perubahan visual. Kedua dataset dilatih menggunakan YOLOv8s selama 30 epoch. Hasil evaluasi menunjukkan bahwa model yang dilatih pada data original memberikan hasil terbaik dengan mAP@0.5 sebesar 0,795 dan mAP@0.5:0.95 sebesar 0,606, sementara model yang dilatih pada data preprocessing hanya mencapai mAP@0.5 sebesar 0,748 dan mAP@0.5:0.95 sebesar 0,560.

Kata kunci: character detection, handwriting, Image processing, preprocessing, YOLOv8

Abstract

Handwritten character recognition is one of the challenges in the field of digital image processing, especially in academic documents such as thesis guidance minutes. This study aims to compare the performance of the YOLOv8 model in detecting handwritten characters on two types of datasets, namely original images and preprocessed images. Preprocessing is carried out through the stages of grayscale, CLAHE, Gaussian blur, adaptive thresholding Gaussian, dilation, and erosion. Labels in the preprocessed data are obtained by copying annotations from the original data without adjusting for visual changes. Both datasets were trained using YOLOv8s for 30 epochs. The evaluation results show that the model trained on the original data gives the best results with mAP@0.5 of 0.795 and mAP@0.5:0.95 of 0.606, while the model trained on the preprocessed data only achieves mAP@0.5 of 0.748 and mAP@0.5:0.95 of 0.560.

Keywords: character detection, handwriting, Image processing, preprocessing, YOLOv8

PENDAHULUAN

Dokumen Berita Acara Bimbingan Skripsi (DBABS) merupakan lembaran kertas berisi catatan tulisan tangan tentang Skripsi atau Tugas Akhir mahasiswa, termasuk proses bimbingan mahasiswa dengan dosen pembimbing

Skripsi. Dokumen ini sering kali diisi secara manual oleh mahasiswa dan dosen pembimbing. Pengecekan DBABS dilakukan untuk

mengetahui apakah dosen pembimbing telah memberikan persetujuan kepada mahasiswa untuk mengikuti Sidang Skripsi. Jumlah mahasiswa yang banyak dan waktu yang terbatas menjadikan pengecekan DBABS secara manual oleh petugas administratif menjadi tugas yang melelahkan.

Tersedianya sistem yang memungkinkan pengecekan DBABS secara otomatis diharapkan dapat membantu dan mempersingkat

proses administratif. Dalam konteks administrasi di perguruan tinggi, digitalisasi DBABS penting dilakukan untuk mendukung pengarsipan. Penelitian ini mengusulkan pengembangan model pengenalan karakter tulisan tangan pada citra dokumen hasil digitalisasi DBABS sebagai langkah awal pembangunan sistem pengecekan DBABS otomatis. Area dokumen yang dikenali mencakup nrp dan nama mahasiswa, judul skripsi, tanggal mulai bimbingan, dan nama dosen pembimbing.

Tulisan tangan setiap individu sangat bervariasi, mulai dari ukuran huruf, kemiringan huruf, tekanan penulisan, jarak antar huruf, bahkan jarak antar kata dalam tulisan. Dokumen tulisan tangan sering kali terpapar berbagai *noise* atau gangguan seperti noda tip-ex, tulisan tumpang tindih, ataupun coretan. Selain itu, setiap individu pada waktu yang berbeda bisa jadi akan menghasilkan gaya tulisan tangan yang berbeda pula. Hal ini merupakan tantangan dalam pengembangan model pengenalan karakter tulisan tangan pada citra dokumen.

Seiring berkembangnya arsitektur *deep learning*, metode deteksi objek seperti *You Only Look Once (YOLO)* telah menunjukkan kinerja unggul dalam berbagai tugas pengenalan seperti pada lontar_detc (Suciati et al., 2022), tajweed (Azizah, 2023) dan digitnet (Kusetogullari et al., 2021). *YOLOv8* diluncurkan pada tanggal 10 Januari 2023 oleh *Ultralytics* yang dirancang untuk deteksi objek, segmentasi, dan klasifikasi. Meskipun telah banyak digunakan untuk mendeteksi objek umum seperti kendaraan atau wajah, namun penerapan *YOLOv8* untuk pengenalan karakter tulisan tangan masih relatif jarang dieksplorasi.

Penelitian ini mengusulkan pendekatan pengenalan karakter tulisan tangan dengan melatih ulang model *YOLOv8* menggunakan dataset citra DBABS yang telah dianotasi pada level karakter. Teknik *preprocessing* citra meliputi normalisasi, *reduksi noise*, peningkatan kontras, dan morfologi (Mahadevkar et al., 2024), dilakukan untuk memperbaiki kualitas citra dokumen. Perbandingan kinerja

hasil deteksi antara model yang dilatih menggunakan data original dengan model yang dilatih menggunakan data hasil praproses dilakukan untuk mengetahui pengaruh *preprocessing* terhadap hasil deteksi. Penilaian kinerja dilakukan menggunakan metrik *Character Error Rate (CER)* dan *Word Error Rate (WER)* (Geetha et al., 2021; Mahadevkar et al., 2024).

TINJAUAN PUSTAKA

Pada bagian ini diuraikan penelitian terdahulu terkait dengan pengenalan tulisan tangan, penerapan teknik *preprocessing* pada dokumen, dan deteksi objek menggunakan *YOLOv8 (You Only Look Once version 8)*, yang menjadi rujukan dalam penelitian ini.

2.1 Pengenalan Karakter Tulisan Tangan

Pengenalan karakter tulisan tangan merupakan sub-bidang visi komputer yang bertujuan mengkonversi citra tulisan tangan menjadi teks digital agar lebih mudah dibaca dan diolah. Berbagai dataset telah digunakan dalam penelitian, seperti IAM dataset untuk bahasa Inggris (Marti & Bunke, 2003), DIDA dataset berisi digit tulisan tangan dari dokumen sejarah Swedia abad ke-19 (Kusetogullari et al., 2021), HBCL dataset untuk aksara Bali pada lontar (Sutramiani et al., 2021), resep dokter (Kamalanaban et al., 2018), serta kumpulan surat bahasa Prancis pada RIMES dataset. Setiap dataset memiliki karakteristik unik, misalnya IAM mencakup lebih dari 10.000 kosakata yang ditulis 432 orang, sedangkan HBCL memerlukan augmentasi data untuk mengatasi keterbatasan jumlah naskah.

Penelitian ini menggunakan dataset DBABS (Dokumen Berita Acara Bimbingan Skripsi) yang dikumpulkan secara mandiri melalui hasil pemindaian maupun foto, terdiri dari 432 citra tulisan tangan dari 432 orang berbeda, kemudian dipotong menjadi beberapa bagian penting (nama mahasiswa, NRP, judul, tanggal, dan nama dosen). Seluruh citra hasil *cropping* selanjutnya diproses melalui tahapan *preprocessing*, lalu dilakukan anotasi pada tiap karakter sebagai data latih

deteksi menggunakan YOLO. Tantangan utama pengenalan tulisan tangan terletak pada variasi gaya individu, *noise*, serta latar belakang dokumen yang tidak seragam, sehingga berbeda dengan pengenalan teks cetak yang lebih terstruktur.

2.2 Preprocessing

Preprocessing merupakan langkah awal yang penting dalam meningkatkan kualitas gambar, terutama dari hasil pemindaian atau kamera seringkali mengandung *noise*, pencahayaan tidak merata, atau ukuran yang tidak konsisten. Pada dataset DBABS, *preprocessing* diperlukan untuk mengurangi gangguan, meningkatkan kontras, dan memperjelas tulisan agar deteksi karakter lebih optimal. Menurut (Mahadevkar et al., 2024), tahap ini dapat mempercepat pengenalan teks. Umumnya, *preprocessing* melibatkan transformasi citra berwarna ke skala abu-abu, binerisasi, serta operasi morfologi seperti dilasi, yang banyak digunakan dalam analisis dokumen dan pengenalan tulisan tangan. Ada beberapa tahap dalam proses *preprocessing*, diantaranya:

2.3 Citra Grayscale

Proses *grayscale* atau mengubah citra berwarna (RGB) menjadi keabuan (*grayscale*) adalah proses untuk menjadikan intensitas pada setiap *pixel* bernilai 0 (hitam) hingga 255 (putih). Gambar yang diubah menjadi citra *grayscale* dapat menyederhanakan pemrosesan gambar untuk tugas deteksi agar lebih fokus pada bentuk dan stuktur objek. Penggunaan citra *grayscale* tidak membutuhkan banyak memori karena hanya memiliki satu *channel*, sehingga mengurangi beban komputasi. Rumus untuk mengubah gambar RGB ke citra skala abu-abu dapat dilihat pada Persamaan 1.

$$Y = 0,299 * R + 0,587 * G + 0,114 * B \quad (1)$$

Y menunjukkan intensitas skala abu-abu, sementara R, G, B masing-masing menunjukkan nilai intensitas saluran warna merah, hijau, dan biru dari gambar asli (Varshini et al., 2024).

2.4 CLAHE (Contrast Limited Adaptive Histogram Equalization)

CLAHE adalah metode peningkatan kontras pada citra digital, yang merupakan pengembangan dari *Adaptive Histogram Equalization* (AHE). AHE mengatasi masalah ini dengan memperbaiki kontras di wilayah lokal. Namun sering menghasilkan *noise* yang tidak diinginkan diarea dengan intensitas rendah. CLAHE membatasi peningkatan kontras dengan menetapkan *clip limit* untuk histogram lokal. Metode ini untuk mempertahankan detail lokal dan mengurangi efek *noise* yang tidak diinginkan muncul. Cara kerja CLAHE adalah dengan membagi citra menjadi blok-blok kecil (disebut *tiles*), misal 8 x 8, atau 16 x 16. *Histogram* intensitas dihitung untuk setiap blok lokal. *Histogram* lokal dipotong hingga mencapai nilai maksimum tertentu (*clip limit*) untuk mencegah peningkatan kontras yang berlebihan. Intensitas yang melampaui *clip limit* didistribusikan secara merata ke seluruh *histogram*. *Histogram* lokal yang telah di-*clip* dinormalisasi untuk merata-takan distribusi intensitas di blok tersebut.

2.5 Gaussian Blur

Noise dalam gambar adalah elemen yang muncul akibat faktor seperti kondisi pencahayaan, kualitas kamera, atau metode akuisisi gambar (Sutramiani et al., 2021). Membersihkan *noise* merupakan salah satu proses dalam *preprocessing* untuk menghilangkan gangguan / *noise* pada gambar. Sehingga meningkatkan kualitas gambar untuk mengenali objek pada citra yang bersih. *Noise* yang tidak dibersihkan akan mengakibatkan kesalahan dalam memprediksi bentuk karakter tulisan tangan.

Salah satu teknik yang digunakan untuk mengurangi *noise* pada gambar adalah dengan menggunakan *gaussian blur*. *Gaussian blur* bekerja dengan cara menghaluskan intensitas piksel menggunakan kernel berbasis distribusi *gaussian*. Kernel adalah matrik kecil dengan nilai yang dihitung berdasarkan fungsi *gaussian*. Sedangkan distribusi *gaussian* untuk menghitung bobot pada setiap piksel di sekitar piksel pusat berdasarkan jaraknya. Piksel yang lebih dekat ke pusat diberi bobot lebih

besar, sementara piksel yang lebih jauh diberi bobot lebih kecil. Rumus menghitung *kernel gaussian* dapat dilihat pada Persamaan 2.

$$G_{(x,y)} = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2)$$

Keterangan :

$G_{(x,y)}$: Nilai bobot yang digunakan untuk piksel tertentu.

(x,y) : Koordinat piksel relatif terhadap pusat *kernel*.

σ : Standar deviasi dari distribusi *Gaussian* untuk menentukan intensitas *blur*.

Kernel biasanya berbentuk persegi dengan dimensi ganjil misalnya 3x3, 5x5, 7x7, dsb. Ukuran *kernel* yang semakin besar akan membuat gambar semakin *blur*. Sedangkan nilai minimal standart deviasi adalah 0 dan jika semakin besar menghasilkan pengaburan lebih kuat. Hasil perkalian nilai piksel pada gambar dengan bobot dijumlahkan, lalu dinormalisasi seperti Persamaan 3.

$$\text{Nilai baru} = \frac{\sum(\text{Nilai Piksel}) \times (\text{Bobot Kernel})}{\sum(\text{Bobot Kernel})} \quad (3)$$

2.6 Adaptive Thresholding

Adaptive Thresholding: nilai *threshold* dihitung secara lokal berdasarkan karakteristik setiap wilayah citra, berguna untuk citra dengan pencahayaan yang tidak merata. *Adaptive mean thresholding* dan *adaptive gaussian thresholding* merupakan metode utama dari *adaptive thresholding*. *Adaptive mean thresholding* untuk mendapatkan nilai ambang dihitung sebagai rata-rata nilai piksel dalam wilayah lokal. Sedangkan *adaptive gaussian thresholding* untuk mendapatkan nilai ambang dihitung sebagai rata-rata berbobot dengan bobot ditentukan oleh *kernel gaussian*. Setiap piksel dibandingkan dengan *threshold local* dihitung pada Persamaan 4.

$$I_{\text{biner}}(x,y) = \begin{cases} 255 & \text{jika } I(x,y) > \text{Threshold Lokal} \\ 0 & \text{lainnya} \end{cases} \quad (4)$$

Parameter utama dalam *adaptive thresholding* yaitu *blockSize* merupakan ukuran kernel lokal untuk menghitung *threshold* dan *C*

merupakan konstanta yang dikurangi dari *threshold* lokal untuk mengontrol sensitivitas.

Dilation

Dilasi adalah operasi *morfologi* dalam pengolahan citra untuk memperluas area objek tertentu dalam citra *biner* (atau *grayscale*), sehingga dapat menghubungkan bagian piksel yang terputus. Penerapan proses ini akan menguatkan kontur dan tepi karakter (Mahadevkar et al., 2024). Proses ini dilakukan dengan mengaplikasikan struktur elemen (*structuring element*) pada citra sehingga area putih (*foreground*) diperluas, dan lubang kecil di dalam objek dapat ditutup. Bentuk struktur elemen (*kernel*) bisa berupa persegi panjang, *elips*, atau *cross*. Ukuran *kernel* memengaruhi seberapa besar area yang diperluas. Jika minimal ada satu piksel *foreground* (putih) di dalam jendela *kernel*, piksel diposisi pusat akan menjadi *foreground*. Area *foreground* diperluas sesuai bentuk dan ukuran kernel.

2.7 Erosion

Erosi adalah operasi dasar *morfologi* citra digital yang digunakan untuk memproses gambar biner. Cara kerja erosi adalah dengan mengikis atau mengurangi tepi pada suatu objek dalam sebuah gambar. Rumus operasi erosi ditunjukkan pada Persamaan 5.

$$A \ominus B = \{z \in E \mid B_z \subseteq A\} \quad (5)$$

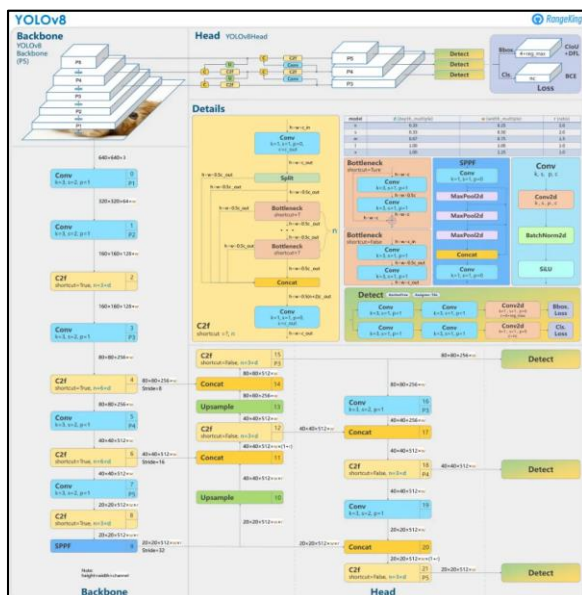
dimana B_z adalah translasi dari elemen struktural B ke posisi z , dan $A \ominus B$ menghasilkan himpunan titik-titik di mana B sepenuhnya berada di dalam A .

Dalam pengenalan karakter tulisan tangan, erosi sering digunakan dalam tahap pra-pemrosesan (*preprocessing*) untuk menghilangkan *noise* atau pengganggu di sekitar karakter, memisahkan karakter yang saling menempel, menajamkan bentuk karakter sebelum dikenali oleh model seperti YOLOv8.

2.8 You Only Look Once (YOLO)

YOLO (*You Only Look Once*) adalah algoritma deteksi objek berbasis CNN yang dirancang untuk mendeteksi lokasi dan

mengklasifikasi berbagai objek dalam gambar atau video dalam satu tahap pemrosesan (Redmon et al., 2016). Pendekatan deteksi objek yang lain menggunakan dua tahap pemrosesan, seperti DPM (*Deformable Parts Models*) dan R-CNN. YOLO membagi gambar menjadi *grid* dan memprediksi kotak pembatas (*bounding boxes*) beserta kelas objek secara bersamaan. Ultralytics telah merilis beberapa versi YOLO, dengan YOLOv8 sebagai versi terbaru yang diluncurkan pada Januari 2023. YOLOv8 dirancang tidak hanya untuk deteksi objek, tetapi juga segmentasi dan tugas visi komputer lain, dengan peningkatan akurasi, kecepatan, dan kemudahan implementasi dibanding pendahulunya. Arsitektur YOLOv8 terdiri dari tiga bagian: *backbone* untuk ekstraksi fitur (*Conv*, *C2f*, dan *SPPF*), *neck* untuk menggabungkan fitur multi-skala (menggunakan FPN dan PANet), serta *head* yang menghasilkan prediksi *bounding box*, *confidence*, dan kelas objek dengan pendekatan *anchor-free*. Resolusi prediksi yang digunakan (P3, P4, P5) membuat YOLOv8 mampu mengenali objek kecil, sedang, hingga besar secara lebih optimal. Detail arsitektur YOLOv8 dapat dilihat pada Gambar 1.

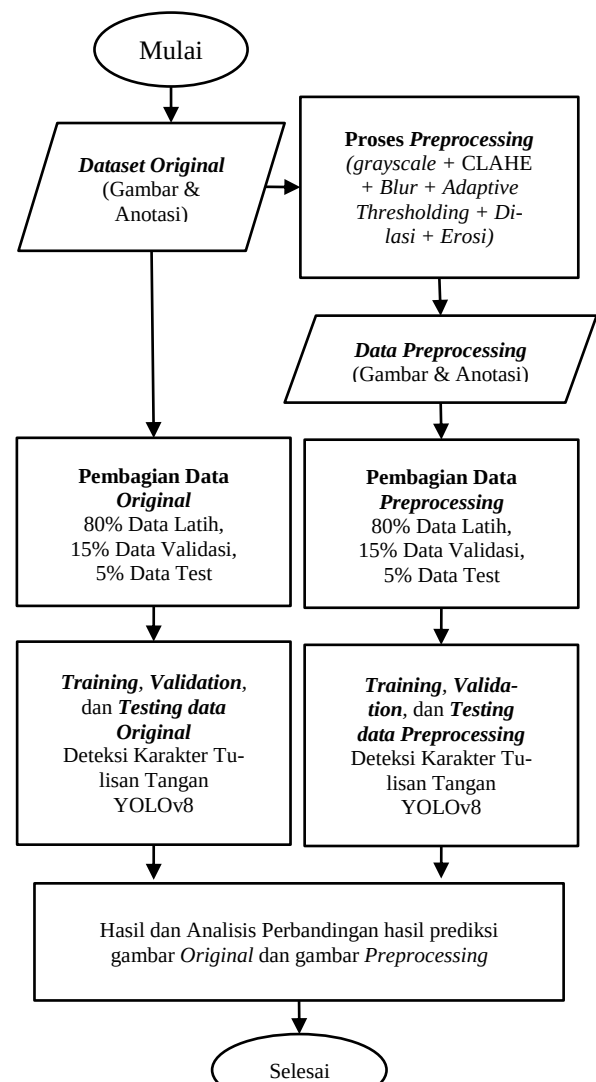


Gambar 1 Arsitektur YOLOv8 (Solawetz & Francesco, 2023)

METODE

Penelitian ini dilakukan melalui beberapa tahap evaluasi. Pertama, karakter tulisan tangan *original* yang telah dianotasi, akan dibagi

menjadi data latih, validasi dan data uji yang digunakan untuk melatih model YOLOv8. Kemudian data *preprocessing* diambil dari gambar *original* yang telah melalui beberapa tahap teknik pengolahan gambar dan dibagi seperti gambar *original*. Kedua dataset akan dibandingkan hasil kinerjanya dan dijelaskan pada bagian hasil dan pembahasan. Gambar 2 menunjukkan keseluruhan proses penelitian ini.



Gambar 2. Alur Penelitian

3.1 Pengumpulan Dataset dan Anotasi

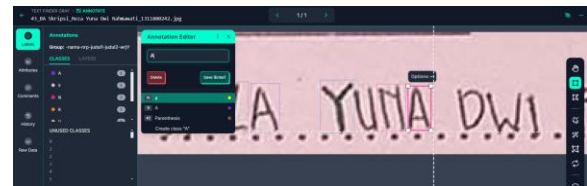
Dataset dikumpulkan dari hasil foto dan *scanning* Dokumen Berita Acara Bimbingan Skripsi (DBABS) yang telah diisi oleh mahasiswa berupa informasi terkait bimbingan seperti nama, nrp, judul tugas akhir, dll, dalam bentuk tulisan tangan. Beberapa berita acara juga disertai isian dosen pembimbing.

Jumlah citra dokumen yang berhasil dikumpulkan adalah 432 citra utuh yang ditulis oleh 432 orang yang berbeda. Citra dokumen yang terkumpul ini memiliki resolusi yang berbeda-beda. Selanjutnya, resolusi semua citra disamakan menjadi 1238 x 1755 piksel menggunakan *photoshop*. Contoh citra dokumen ditunjukkan pada Gambar 4.

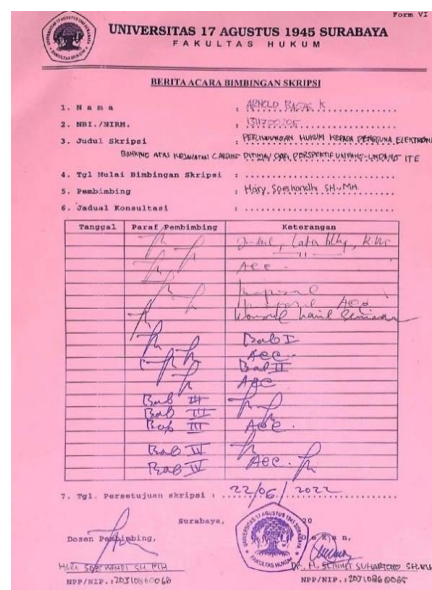
Untuk mengurangi beban komputasi dan penggunaan memori pada saat pelatihan model pengenalan karakter tulisan tangan, dilakukan pemotongan citra (*cropping*) pada bagian citra yang menjadi target pengenalan, yaitu pada area nama mahasiswa, NRP, judul, tanggal mulai bimbingan, dan nama dosen pembimbing. Area dan titik koordinat DBABS ditunjukkan pada Tabel 1. Proses pemotongan menghasilkan citra berisi tulisan tangan dengan ukuran 638 piksel dan 1088 piksel,

dengan total 2.395 citra.

Selanjutnya, dilakukan proses anotasi pada citra hasil pemotongan menggunakan Roboflow. Proses anotasi menghasilkan 67 *class*, yang terdiri dari 26 huruf alfabet kapital (A-Z), 26 huruf alfabet kecil (a-z), angka (0-9), dan simbol (‘(’, ‘)’, ‘&’, ‘-’, ‘/’). Tabel 2 menunjukkan hasil anotasi berupa ID, label, jumlah citra, contoh citra dari 67 *class* karakter tulisan tangan yang digunakan pada DBABS.



Gambar 3. Proses Annotation dan Labelling pada Citra Asli



Gambar 4. Contoh citra DBABS

Tabel 1. Area dan koordinat DBABS yang dipotong

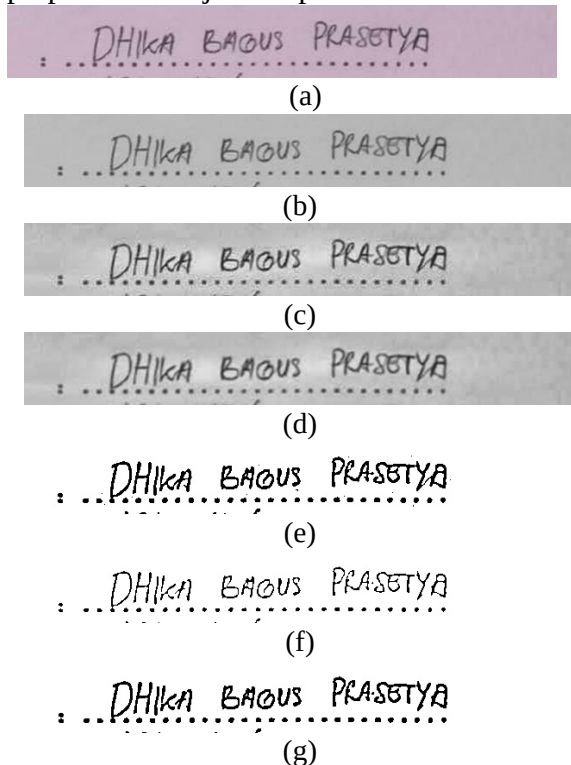
No.	Bagian yang dipotong	Koordinat (x,y,w,h)	Hasil Crop
1.	Nama mahasiswa	600, 240, 1238, 80	Achmad Haikal Robbani
2.	NBI. / NIRM.	600, 295, 1238, 80	1311700141
3.	Judul Skripsi 1	600, 350, 1238, 80	Perlindungan Hukum pemegang hak Merek yang Berlabel sebagai wakil Dalam Perwakilan bertamaka
4.	Judul Skripsi 2	150, 398, 1238, 95	Berlabel sebagai wakil Dalam Perwakilan bertamaka Waktu 14 April 2021
5.	Tanggal Mulai Bimbingan Skripsi	600, 440, 1238, 80	14 April 2021
6.	Pembimbing	600, 500, 1238, 80	Moh. Jupri Ahmad, SH., MM., MH.

Tabel 2. Kelas dan jumlah karakter serta contoh citra tulisan tangan yang telah dianotasi

ID	Label	Jumlah	Contoh Citra	ID	Label	Jumlah	Contoh Citra	ID	Label	Jumlah	Contoh Citra
0	-	137		23	K	444		46	g	472	
1	0	1.545		24	L	318		47	h	567	
2	1	1.778		25	M	1.012		48	i	1.686	
3	2	1.097		26	N	769		49	j	121	
4	3	565		27	O	354		50	k	517	
5	4	119		28	) (OP)	15		51	l	513	
6	5	145		29	P	733		52	m	723	
7	6	142		30	Q	9		53	n	1.639	
8	7	192		31	R	618		54	o	674	
9	8	276		32	S	1.140		55	p	366	
10	9	282		33	T	540		56	q	8	
11	A	1.687		34	U	418		57	r	1.343	
12	&	1		35	V	35		58	s	656	
13	B	238		36	W	181		59	/	37	
14	C	141		37	X	1		60	t	858	
15	CP	16		38	Y	186		61	u	932	
16	D	616		39	Z	33		62	v	67	
17	E	594		40	a	2.977		63	w	126	
18	F	175		41	b	336		64	x	3	
19	G	214		42	c	72		65	y	274	
20	H	1.137		43	d	678		66	z	33	
21	I	682		44	e	1.554					
22	J	131		45	f	166					

3.2 Preprocessing

Teknik *preprocessing* bertujuan untuk meningkatkan kualitas citra masukan agar proses deteksi dan pengenalan karakter lebih akurat (Mahadevkar et al., 2024; Suciati et al., 2022). Dalam penelitian ini *preprocessing* dilakukan melalui beberapa tahapan yang mencakup konversi citra berwarna ke *grayscale*, penajaman citra menggunakan CLAHE, pengaburan citra menggunakan fungsi *Gaussian*, konversi ke citra hitam putih menggunakan *Adaptive Thresholding Gaussian*, operasi morfologi dilasi dan erosi. Intensitas piksel, pengurangan *noise*, peningkatan kontras, binarisasi, dan operasi morfologi. Contoh *output* citra pada setiap tahapan praproses ditunjukkan pada Gambar 5.



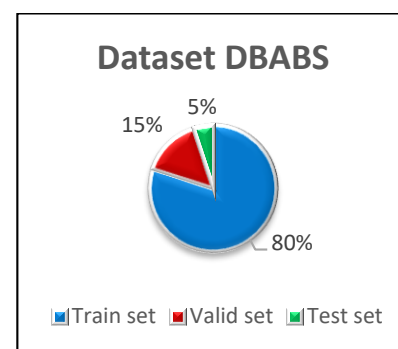
Gambar 5. *Output* dari tahapan praproses (a) citra RGB, (b) citra *Grayscale*, (c) CLAHE, (d) *Gaussian Blur*, (e) *Adaptive Threshold*, (f) Dilasi, (g) Erosi.

3.3 Deteksi Karakter Tulisan Tangan

Setelah tahapan *preprocessing*, proses deteksi huruf dilakukan menggunakan model YOLOv8 (You Only Look Once versi 8), yang merupakan salah satu arsitektur deteksi objek dalam mendeteksi banyak objek kecil secara cepat dan akurat (Azizah, 2023; Kamboj et al., 2024; Kusetogullari et al., 2021; Suciati et al.,

2022). Model yang digunakan adalah varian YOLOv8s (small), yang dipilih karena mampu memberikan keseimbangan optimal antara efisiensi komputasi dan akurasi deteksi pada citra tulisan tangan.

Model dilatih pada dataset karakter hasil anotasi manual dengan format YOLO, yang mencakup huruf kapital dan kecil, angka, serta beberapa simbol dasar. Pembagian jumlah *images* pada dataset DBABS gambar original maupun praproses ditunjukkan pada Gambar 6. Masing-masing terdiri dari 1.480 *images* data *Train*, 278 *images* data *Validation*, dan 92 *images* data *Test*.



Gambar 6. Pembagian dataset DBABS

Pelatihan dilakukan 30 *epochs*, dengan konfigurasi ukuran citra 640×640 piksel, *batch size* 16, dan menggunakan AdamW optimizer dengan *learning rate* otomatis. Model disimpan pada bobot terbaik (*best.pt*) berdasarkan metrik *mAP50-95* tertinggi pada data validasi.

Setiap citra diuji menggunakan model YOLOv8 yang telah dilatih. *Output* berupa koordinat *bounding box*, label kelas karakter, dan skor keyakinan. Hasil deteksi ini kemudian disusun ulang berdasarkan posisi spasial karakter untuk membentuk kata atau kalimat menggunakan pendekatan pengelompokan baris dan deteksi spasi antar karakter. Penyusunan teks dilakukan dengan mempertimbangkan tinggi rata-rata karakter sebagai batas pemisah antar baris, serta jarak antar *bounding box* sebagai indikator spasi antar kata.

Secara keseluruhan, pendekatan ini memanfaatkan kekuatan YOLOv8 dalam deteksi karakter individual, yang kemudian disusun

ulang secara kontekstual menjadi kalimat sesuai tata letak pada dokumen bimbingan skripsi.

3.4 Evaluasi Kinerja Deteksi

Evaluasi kinerja sistem dilakukan dengan dua pendekatan utama, yaitu evaluasi berbasis deteksi objek (*bounding box*) dan evaluasi berbasis hasil pembentukan teks dari prediksi karakter.

3.5 Evaluasi Deteksi Karakter

Evaluasi ini bertujuan mengukur kemampuan model dalam mendeteksi dan mengklasifikasi setiap karakter huruf secara individual. Pengukuran dilakukan dengan metrik yang umum digunakan dalam domain *object detection*, yaitu:

Precision (P) : persentase prediksi karakter yang benar dari seluruh prediksi model, dapat dilihat pada Persamaan 6.

$$precision = \frac{TP}{TP+FP} \quad (6)$$

Recall (R) : persentase karakter yang benar-benar terdeteksi dari seluruh karakter yang ada di *ground truth*, dapat dilihat pada Persamaan 7.

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

mAP@0.5 (*mean Average Precision*): rata-rata presisi pada *threshold Intersection over Union* (*IoU*) sebesar 0.5, yang menunjukkan seberapa tepat model dalam memprediksi posisi dan label, dapat dilihat pada Persamaan 8.

$$mAP = \frac{1}{N} \sum_{k=0}^{k=N} AP_k \quad (8)$$

mAP@0.5:0.95: rata-rata presisi pada berbagai nilai *IoU* (mulai dari 0.5 hingga 0.95 dengan langkah 0.05), yang lebih ketat dan mencerminkan keakuratan spasial prediksi.

Dalam Persamaan (6-7), TP adalah hasil dari kelas positif yang diprediksi dengan benar, FP adalah hasil dari kelas positif yang diprediksi secara salah, dan FN adalah hasil dari kelas negatif yang diprediksi secara salah. Persamaan (8) adalah jumlah kueri. Evaluasi

kinerja model deteksi dilakukan pada *Ultra-lytics* YOLOv8.2.103 Python-3.12.4 torch-2.2.2+cu121 CUDA:0 (NVIDIA GeForce RTX 3050 Laptop GPU, 4096MiB), Setup complete (12 CPUs, 15.7 GB RAM, 193.8/268.1 GB disk). Jumlah waktu untuk melatih model adalah 6 jam 8 menit pada dataset original dan 1 jam 12 menit pada dataset *preprocessing*.

3.6 Evaluasi Pembentukan Teks

Karakter-karakter yang telah terdeteksi oleh model akan direkonstruksi menjadi kata dan kalimat menggunakan pendekatan berbasis posisi *bounding box* (koordinat x dan y). Evaluasi keberhasilan pembentukan kata atau kalimat dilakukan dengan membandingkan hasil prediksi terhadap *ground truth* teks yang disediakan dalam file referensi (.csv).

Dua metrik utama digunakan (Geetha et al., 2021; Mahadevkar et al., 2024)

Character Error Rate (*CER*) : untuk mengukur seberapa banyak karakter yang salah dalam hasil prediksi dibandingkan dengan *ground truth*. Menghitung *CER* dapat menggunakan Persamaan 9.

$$CER = \frac{S+D+I}{N} \quad (9)$$

dengan:

S = jumlah substitusi,

D = jumlah karakter yang hilang,

I = jumlah karakter yang salah ditambahkan,

N = jumlah total karakter *ground truth*.

Perbandingan dilakukan secara langsung antar file hasil prediksi dan *ground truth* berdasarkan nama file gambar. Proses evaluasi ini membantu mengukur sejauh mana hasil deteksi karakter mampu dikonversi menjadi informasi teks yang bermakna.

HASIL DAN PEMBAHASAN

4.1 Dataset

Dataset yang digunakan adalah HCR_original_image 2 yang terdiri dari 1.850 gambar beserta 36.114 karakter berlabel, yang dibangun melalui proses pembuatan data menggunakan gambar dokumen berita acara bimbingan skripsi utuh. Kemudian dipotong

pada area seperti yang ditunjukkan pada Tabel 1 dan dianotasi seperti pada Gambar 3. Kedua dataset tersebut dibandingkan hasil kinerja model yang telah dilatih, validasi, dan test.

4.2 Hasil Deteksi Karakter Huruf pada Data Original

Pelatihan dilakukan 30 *epochs* selama 6.135 *hours*, dengan konfigurasi ukuran citra 640×640 piksel, *batch size* 16, dan *learning rate* otomatis. Model disimpan pada bobot terbaik (best.pt) berdasarkan metrik *mAP50-95*

tertinggi pada data validasi. Tabel 3 menampilkan hasil evaluasi per kelas pada data test *original* dan *preprocessing*. Beberapa label tampak bermasalah seperti label simbol “&” memperoleh *mAP* 0, yang berarti tidak terdeteksi sama sekali. Label simbol “), G, c, j, “slash” dan “z” mendapatkan *mAP* sangat rendah, karena kurang dari 5 contoh dalam dataset. Label yang tidak memiliki contoh dalam data test adalah simbol “(“ OP, Q, V, X, q, x. Dataset yang tidak seimbang membuat model tidak terlalu mampu belajar cukup dari data langka.

Tabel 3. Hasil Evaluasi Per Kelas pada Data Test *Original* dan Data *Preprocessing*

Label	Jumlah gambar	Evaluasi Pada Data Original				Evaluasi Pada Data <i>Preprocessing</i>			
		Precision	Recall	mAP@0.5	mAP@0.5 :0.95	Precision	Recall	mAP@0.5	mAP@0.5 :0.95
- (dash)	6	0,796	0,436	0,532	0,188	0,959	0,667	0,682	0,302
0	35	1	0,971	0,989	0,796	0,987	0,958	0,984	0,775
1	25	0,96	0,941	0,983	0,656	0,923	0,909	0,966	0,645
2	21	0,931	0,944	0,982	0,766	0,961	0,925	0,975	0,731
3	21	1	0,913	0,993	0,865	0,98	0,929	0,986	0,828
4	5	0,869	0,6	0,637	0,49	1	0,553	0,676	0,527
5	4	0,539	1	0,995	0,834	0,647	1	0,845	0,702
6	10	1	0,694	0,808	0,607	0,969	0,8	0,808	0,605
7	5	0,873	1	0,995	0,774	0,86	1	0,995	0,728
8	12	0,871	0,929	0,958	0,823	0,849	0,929	0,929	0,797
9	12	0,841	1	0,986	0,832	0,889	0,857	0,948	0,826
A	22	0,884	0,753	0,903	0,626	0,865	0,705	0,841	0,558
& (Ampersand)	1	0	0	0	0	0	0	0	0
B	7	0,682	0,75	0,88	0,735	0,713	0,75	0,756	0,584
C	5	0,813	0,5	0,698	0,617	0,695	0,667	0,793	0,685
) CP	1	0,909	1	0,995	0,697	1	0	0,142	0,114
D	27	0,825	0,829	0,939	0,768	0,859	0,743	0,895	0,715
E	14	0,891	0,683	0,896	0,588	0,928	0,538	0,679	0,462
F	8	0,768	1	0,94	0,776	0,879	0,875	0,944	0,749
G	3	0,779	0,6	0,665	0,455	0,509	0,2	0,399	0,323

Label	Jumlah gambar	Evaluasi Pada Data Original				Evaluasi Pada Data <i>Preprocessing</i>			
		Precision	Recall	mAP@0.5	mAP@0.5 :0.95	Precision	Recall	mAP@0.5	mAP@0.5 :0.95
H	32	0,95	0,914	0,975	0,822	0,981	0,909	0,947	0,792
I	17	0,779	0,333	0,598	0,403	0,94	0,519	0,594	0,383
J	5	0,545	0,571	0,607	0,519	0,713	0,571	0,595	0,443
K	12	0,631	0,429	0,492	0,364	0,476	0,312	0,388	0,292
L	10	0,88	0,492	0,707	0,484	0,706	0,4	0,633	0,354
M	36	0,846	0,837	0,924	0,76	0,872	0,878	0,936	0,748
N	14	0,691	0,571	0,704	0,52	0,813	0,536	0,667	0,498
O	12	1	0,651	0,848	0,684	0,927	0,609	0,763	0,594
(OP	-	-	-	-	-	-	-	-	-
P	21	0,775	0,897	0,915	0,721	0,878	0,966	0,96	0,707
Q	-	-	-	-	-	-	-	-	-
R	17	0,781	0,528	0,617	0,453	0,758	0,593	0,624	0,472
S	43	0,898	0,803	0,899	0,73	0,949	0,839	0,902	0,707
T	16	0,682	0,684	0,751	0,548	0,594	0,5	0,607	0,407
U	13	0,549	0,647	0,673	0,523	0,835	0,594	0,749	0,541
V	-	-	-	-	-	-	-	-	-
W	8	0,759	0,792	0,92	0,791	0,883	0,875	0,879	0,803
X	-	-	-	-	-	-	-	-	-
Y	10	1	0,631	0,814	0,638	1	0,516	0,85	0,625
Z	1	0,523	1	0,995	0,796	0,562	1	0,995	0,697
a	48	0,78	0,785	0,831	0,596	0,809	0,765	0,839	0,578
b	17	0,847	0,889	0,918	0,702	0,931	0,745	0,922	0,72
c	3	0,453	0,667	0,521	0,423	0,623	0,579	0,446	0,357
d	28	0,755	0,75	0,845	0,656	0,65	0,757	0,759	0,576
e	36	0,865	0,705	0,825	0,567	0,873	0,707	0,807	0,535
f	11	0,805	0,637	0,807	0,556	1	0,512	0,863	0,623
g	18	0,864	0,759	0,887	0,674	0,834	0,724	0,843	0,6
h	25	0,777	0,758	0,839	0,664	0,836	0,697	0,832	0,657
i	41	0,849	0,646	0,765	0,413	0,903	0,646	0,769	0,419
j	4	0,614	0,25	0,3	0,242	0,54	0,31	0,421	0,403
k	16	0,607	0,37	0,506	0,388	0,678	0,391	0,485	0,346

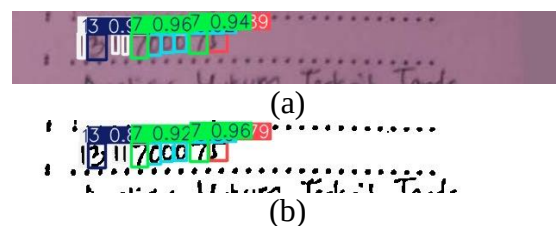
Label	Jumlah gambar	Evaluasi Pada Data Original				Evaluasi Pada Data <i>Preprocessing</i>			
		Precision	Recall	mAP@0.5	mAP@0.5 :0.95	Precision	Recall	mAP@0.5	mAP@0.5 :0.95
l	18	0,583	0,348	0,442	0,287	0,819	0,348	0,454	0,276
m	39	0,876	0,686	0,841	0,613	0,816	0,667	0,789	0,557
n	35	0,78	0,814	0,88	0,631	0,812	0,759	0,869	0,584
o	24	0,856	0,662	0,778	0,598	0,795	0,539	0,729	0,555
p	14	0,906	0,689	0,915	0,619	0,822	0,643	0,831	0,576
q	-	-	-	-	-	-	-	-	-
r	48	0,896	0,723	0,859	0,572	0,878	0,665	0,802	0,535
s	25	0,825	0,611	0,709	0,461	0,73	0,6	0,65	0,408
/ (slash)	2	1	0	0,409	0,245	1	0	0,332	0,172
t	31	0,882	0,8	0,906	0,647	0,946	0,75	0,866	0,572
u	31	0,789	0,8	0,804	0,572	0,817	0,709	0,81	0,556
v	2	0,982	1	0,995	0,587	0,884	1	0,995	0,642
w	6	1	0,832	0,909	0,73	0,607	0,75	0,864	0,651
x	-	-	-	-	-	-	-	-	-
y	13	0,878	0,714	0,828	0,598	0,836	0,786	0,864	0,655
z	2	1	0	0,586	0,527	1	0	0,179	0,161
All	92	0,803	0,692	0,789	0,594	0,816	0,65	0,743	0,548

Secara umum, model cukup baik dalam mendeteksi huruf yang sering muncul (seperti S, a, i, r, dll). Huruf yang jarang muncul (seperti 4, 5, 7, “&”, C, “(”, G, J, Z, c, j, “/” v, z) memiliki performa rendah, menandakan perlunya penambahan data. Evaluasi perkelas penting untuk menganalisis keseimbangan data dan potensi augmentasi data tambahan.

4.3 Hasil Deteksi Karakter Huruf pada Data *Preprocessing*

Pelatihan pada data *preprocessing* dilakukan dengan jumlah *epochs* yang sama, namun prosesnya lebih cepat yaitu selama 1 jam 12 menit. Konfigurasi ukuran citra 640×640 piksel, *batch size* 16, dan *learning rate* otomatis. Hasil deteksi ini kemudian disusun ulang berdasarkan posisi spasial karakter untuk membentuk kata atau kalimat menggunakan pendekatan pengelompokan baris dan deteksi

spasi antar karakter. Penyusunan teks dilakukan dengan mempertimbangkan tinggi rata-rata karakter sebagai batas pemisah antar baris, serta jarak antar *bounding box* sebagai indikator spasi antar kata. Gambar 7 menampilkan hasil deteksi pada citra original dan citra *preprocessing*. Tabel 4 menampilkan perbandingan hasil prediksi antara gambar original dan gambar *preprocessing*.



Gambar 7. Contoh Visualisasi Hasil Deteksi pada citra original (a) dan citra yang sudah di *Preprocess* (b)

Tabel 4 Hasil Deteksi

Ground truth	Prediksi Gambar original	Prediksi Gambar Preprocessing
1311600115	1311 1 6 001 1 5	1311 6001 1 5
Widhi Cahyo Nugroho SH M Hum	Widhi Catty Nugrotho SH M Hum	Widh Cathyo N u graddao S H M Hum
4 Oktober 2020	4 O ktober 2020	A Oo ktober 2020
1311700075	113 1 1 7000 75	1 3 1 1 70 0 0 7 75
16 Maret 2021	16 Maret 2021	16 Ma et 2021
Ariseno Yusuf Halim	Arisena Yusuf Haliimn	Arise a Yusuf Haliimm
Refina Mirza D	Raefiir a Minsa D	ReFina MMN a D
Wiwik Afifah S Pi SH MH	Wiwik Afifah S Pi SH MH	WMiwik Afifah S Pi SH MH
2 September 2020	2 September 2020	2 September 2020
Istifarah Dinda	Iti h Dinda e aroa	Iti h Dindna e ara
1311700010	13 1700010	13 1700010
Perlindungan Hukum Terhadap Konsumen	Perlin dunggan Hu um Terhadap Konsumen	Per i n dungggga n Hukum Terhadap Kons umen
Dr Syofyan Hadi SH MH	Dr Syofyan Hadii S H M H	Dr Syofyan Hadi S H M H
Peran Pemerintah Dalam Meminimalisir	PeranPe-megrintah Dalam MMAenmiiinnal isir	PeranPe-megrintah Dalt-ammnmn Meniin alisSr
Wewenang Dewan Perwakilan	WW eg e nan PDe an Perr wakt a ng	KWWegug-nana DPentanPerrr-watkLan gg

SIMPULAN

Penelitian ini membandingkan performa model YOLOv8s dalam mendeteksi karakter tulisan tangan pada dua jenis dataset, yakni gambar original dan gambar hasil *preprocessing*. *Preprocessing* dilakukan melalui tahapan *grayscale*, *CLAHE*, *Gaussian blur*, *adaptive thresholding Gaussian*, dilasi, dan erosi. Pelatihan dilakukan

selama 30 *epoch* dengan label pada data *preprocessing* merupakan salinan dari data original tanpa penyesuaian ulang terhadap bentuk visual yang berubah. Berdasarkan hasil evaluasi, model yang dilatih pada data original menunjukkan performa yang lebih tinggi dengan nilai mAP@0.5 sebesar 0,795 dan mAP@0.5:0.95 sebesar 0,606. Sementara itu, model pada data *preprocessing* memiliki kinerja lebih rendah, dengan mAP@0.5 sebesar 0,748 dan mAP@0.5:0.95 sebesar 0,560. Hal ini menunjukkan bahwa perubahan visual akibat *preprocessing* tanpa disertai pelabelan ulang dapat menyebabkan penurunan akurasi deteksi.

Pada penelitian selanjutnya, dapat dilakukan beberapa hal untuk meningkatkan kinerja hasil deteksi. Pertama, anotasi ulang data hasil *preprocessing*, agar model dapat beradaptasi dengan representasi visual baru. Kedua, mengeksplorasi penggunaan rangkaian *preprocessing* yang lain serta penggunaan YOLO versi yang lebih baru. Selain itu, penelitian ini perlu dikembangkan agar bisa mengenali semua bagian dari berita acara bimbingan skripsi serta mengenali tanda tangan setiap individu, sehingga proses rekapitulasi bimbingan skripsi dapat dilakukan secara otomatis.

DAFTAR PUSTAKA

- Azizah, A. N. , F. C. (2023). Tajweed-YOLO: Object Detection Method for Tajweed by Applying HSV Color Model Augmentation on Mushaf Images. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 7(2). <https://doi.org/10.29207/resti.v7i2.4739>
- Geetha, R., Thilagam, T., & Padmavathy, T. (2021). Effective offline handwritten text recognition model based on a sequence-to-sequence approach with CNN–RNN networks. In *Neural Computing and Applications* (Vol. 33, Issue 17). <https://doi.org/10.1007/s00521-020-05556-5>
- Kamboj, D., Dipanjali, Mittal, A., Tiwari, A., & Agarwal, D. (2024). DigiSketch: Automating the Conversion of Hand-Drawn Diagrams to Digital. *Proceedings of International Conference on Communication, Computer Sciences and Engineering, IC3SE 2024*, 354–358. <https://doi.org/10.1109/IC3SE62002.2024.10592887>

- Kusetogullari, H., Yavariabdi, A., Hall, J., & Lavesson, N. (2021). DIGITNET: A Deep Handwritten Digit Detection and Recognition Method Using a New Historical Handwritten Digit Dataset. *Big Data Research*, 23. <https://doi.org/10.1016/j.bdr.2020.100182>
- Mahadevkar, S., Patil, S., & Kotecha, K. (2024). Enhancement of handwritten text recognition using AI-based hybrid approach. *MethodsX*, 12. <https://doi.org/10.1016/j.mex.2024.102654>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-December*, 779–788. <https://doi.org/10.1109/CVPR.2016.91>
- Solawetz, J., & Francesco. (2023). *What is YOLOv8? The Ultimate Guide*. Roboflow. <https://blog.roboflow.com/whats-new-in-yolov8/>
- Suciati, N., Sutramiani, N. P., & Siahaan, D. (2022). LONTAR-DETC: Dense and High Variance Balinese Character Detection Method in Lontar Manuscripts. *IEEE Access*, 10, 14600–14609. <https://doi.org/10.1109/ACCESS.2022.3147069>
- Sutramiani, N. P., Suciati, N., & Siahaan, D. (2021). MAT-AGCA: Multi Augmentation Technique on small dataset for Balinese character recognition using Convolutional Neural Network. *ICT Express*, 7(4), 521–529. <https://doi.org/10.1016/j.icte.2021.04.005>
- Varshini, C., Yogeshwaran, S., & Mekala, V. (2024). Tamil and English Handwritten Character Segmentation and Recognition Using Deep Learning. *2024 International Conference on Communication, Computing and Internet of Things, IC3IoT 2024 - Proceedings*. <https://doi.org/10.1109/IC3IoT60841.2024.10550221>