

Pengaruh Encoder Residual pada Kinerja U-Net untuk Binarisasi Manuskrip Historis

Rino Zakharia¹, Budi Nugroho^{2*}, Eka Prakarsa Mandyartha³

^{1,2,3} Informatika, Universitas Pembangunan Nasional “Veteran” Jawa Timur

¹21081010045@student.upnjatim.ac.id

³eka_prakarsa.fik@upnjatim.ac.id

*Corresponding author email: budinugroho.if@upnjatim.ac.id

Abstrak— Binarisasi merupakan tahap penting dalam analisis citra manuskrip historis karena kualitas pemisahan teks dan latar sangat memengaruhi keterbacaan aksara serta proses lanjutan seperti segmentasi dan pengenalan karakter. Penelitian ini menganalisis pengaruh penggunaan *encoder residual* pada kinerja *U-Net* untuk binarisasi manuskrip daun lontar dengan membandingkan tiga model, yaitu *U-Net* dasar, *U-Net ResNet34* tanpa *pretrained*, dan *U-Net ResNet34 pretrained*. Data yang digunakan berasal dari dua dataset, yaitu AMADI Lontarset dan THPLMD, dengan total 261 citra yang dibagi ke dalam data latih, validasi, dan uji dengan rasio 80:10:10. Seluruh citra diproses menggunakan pendekatan *patch-based* berukuran 256×256 piksel. Evaluasi dilakukan menggunakan metrik IoU, *Dice*, *Recall*, *Accuracy*, SSIM, FSIM, MS-SSIM, GMSD, RMSE, dan PSNR, serta aspek efisiensi komputasi. Hasil penelitian menunjukkan bahwa kedua model *U-Net ResNet34* memberikan kinerja lebih baik daripada *U-Net* dasar pada hampir seluruh metrik. *U-Net ResNet34* tanpa *pretrained* memperoleh hasil terbaik pada IoU sebesar 0,7292, *Dice* 0,8266, dan *Recall* 0,8214, sedangkan *U-Net ResNet34 pretrained* unggul pada *Accuracy* sebesar 0,9783, SSIM 0,9192, FSIM 0,9385, RMSE 0,1378, dan PSNR 17,7469. Selain itu, kedua varian *ResNet34* juga lebih efisien dari sisi waktu pelatihan, inferensi, jumlah parameter, dan penggunaan memori. Temuan ini menunjukkan bahwa *encoder residual* berperan penting dalam meningkatkan performa binarisasi, sementara pengaruh *pretrained* bersifat tambahan.

Kata Kunci— binarisasi citra, manuskrip historis, *U-Net*, *ResNet34*, *encoder residual*, pengolahan citra dokumen

I. PENDAHULUAN

Di Indonesia dan penjuru dunia, naskah kuno memiliki nilai historis dan pengetahuan yang tinggi, oleh sebab itu naskah kuno menjadi sumber penting bagi penelitian budaya dan Sejarah [8]. Namun dokumen ini, jarang hadir dalam kondisi yang baik untuk diproses oleh komputer atau mesin. Gangguannya bukan hanya karena usia naskah, tetapi juga karena sifat materialnya. Serat daun, noda, perubahan warna, dan tinta yang memudar. Kemudian karena faktor lingkungan seperti kelembapan, suhu, jamur, dan serangga yang merupakan masalah lain yang dihadapi oleh naskah [9] [12]. Bagi manusia, gangguan seperti ini terkadang masih bisa diatasi lewat konteks bacaan. Namun bagi mesin atau komputer, satu goresan tipis yang salah dibaca sebagai latar belakang dapat membuat bentuk aksara berubah, terpotong, atau hilang. Karena itu binarisasi masih menjadi tahap yang

sangat menentukan dalam analisis citra dokumen. Beberapa metode klasik seperti Otsu dan Sauvola merupakan algoritma yang bisa digunakan untuk melakukan binarisasi [2],[3]. Keduanya tergolong metode klasik yang hingga sekarang digunakan sebagai pembanding terhadap metode-metode baru. Namun kinerjanya tidak selalu stabil ketika degradasi muncul secara tidak merata pada dokumen [1] [12] [13].

Penelitian pada [1] membandingkan sepuluh metode binarisasi klasik dengan *U-net* yang menggunakan *Resnet 34 pre-trained* sebagai *encoder* pada dua dataset manuskrip lontar Bali, yaitu Lontar Terumbalan dan AMADI_Lontarset. Model *deep learning* tersebut dilatih pada blok citra 256×256 dan menghasilkan performa terbaik hampir di seluruh metrik evaluasi menungguli metode klasik. Hasil ini selaras dengan studi-studi binarisasi berbasis *deep learning* yang menunjukkan bahwa model segmentasi piksel mampu mempelajari pola degradasi yang sulit ditangani aturan thresholding tetap [14],[15]. Bahkan, pendekatan berbasis encoder-decoder juga telah dilaporkan efektif pada dokumen manuskrip Nusantara lain [16].

Sebelum pendekatan tersebut, *U-net* sebagai model dasar juga telah digunakan untuk proses binarisasi citra. *U-Net*, yang diperkenalkan sebagai arsitektur *encoder-decoder* dengan *skip connection*, memberi jalan yang lebih baik untuk menjaga informasi spasial selama proses rekonstruksi. Dalam ranah binarisasi dokumen, *U-Net* telah digunakan secara langsung untuk memodelkan tugas ini sebagai klasifikasi piksel teks dan latar. Lebih jauh lagi, model tersebut meraih peringkat pertama pada DIBCO 2017, yang menunjukkan bahwa *U-Net* dasar bukan sekadar *baseline* pelengkap, melainkan model yang memang kompetitif untuk binarisasi [4],[5].

Meski begitu, masih ada satu pertanyaan yang belum benar-benar dijawab oleh penelitian [1]. Model *deep learning* yang dibandingkan penelitian tersebut bukan menggunakan *U-Net* dasar, melainkan dengan metode atau model klasik. Dengan kata lain, keunggulan yang dilaporkan belum sepenuhnya membuktikan pengaruh penggunaan *encoder residual* dengan *encoder* pada model dasar *U-net*. *ResNet* diperkenalkan untuk mempermudah optimisasi jaringan yang lebih dalam melalui *residual learning*. Namun dalam binarisasi dokumen, tambahan kedalaman dan kompleksitas arsitektur seharusnya dibayar dengan peningkatan performa yang benar-benar berarti. Jika selisihnya kecil, maka *U-Net* dasar bisa saja sudah cukup memadai [1],[4],[6]. Selain itu, penelitian [16], [17] menunjukkan bahwa pengembangan binarisasi masih bergerak

ke berbagai arah, misalnya gated U-Net dan Transformer-CNN. Namun, sebelum membandingkan model-model kompleks tersebut, tetap diperlukan evaluasi untuk mengetahui secara khusus sejauh mana *encoder residual* pada U-Net benar-benar berkontribusi terhadap peningkatan kinerja.

Berdasarkan celah tersebut, penelitian ini membandingkan U-Net dasar dan U-Net ResNet34 untuk binarisasi dokumen kuno Indonesia dalam skenario eksperimen yang setara. Perbandingan ini tidak hanya dilakukan terhadap model deep learning, tetapi juga diperkuat dengan baseline klasik Otsu dan Sauvola sebagai pembanding metode *thresholding*. Selain itu, eksperimen dilengkapi dengan data *augmentation* pada data latih, peningkatan jumlah epoch pelatihan, validasi menggunakan 3-fold cross-validation, serta uji statistik untuk mengetahui apakah perbedaan kinerja antar metode memiliki makna secara kuantitatif. Dengan rancangan tersebut, penelitian ini diharapkan dapat memberikan dasar yang lebih kuat dalam menilai kontribusi *encoder residual* pada U-Net, baik dari sisi performa binarisasi maupun efisiensi komputasi pada manuskrip historis.

II. KAJIAN PUSTAKA

A. Pengolahan Citra Digital

Pengolahan citra digital dapat dipahami sebagai rangkaian teknik untuk memperoleh, memperbaiki, mengubah, dan menganalisis citra agar informasi di dalamnya lebih mudah dibaca atau diekstrak. Dalam praktiknya, tahap-tahap yang sering muncul meliputi akuisisi citra, *enhancement*, *restoration*, *segmentation*, *feature extraction*, hingga *classification* [7]. Pada citra dokumen, terutama dokumen kuno, pengolahan citra digital biasanya bergerak dari tahap perbaikan kualitas citra menuju tahap pemisahan objek utama dari latar. Di sini, *enhancement* dipakai untuk memperbaiki kontras atau mengurangi *noise*, sedangkan segmentasi dipakai untuk memisahkan area yang bermakna dari area yang tidak relevan. Binarisasi berada di tengah alur itu. Ia menggunakan hasil perbaikan citra, tetapi sekaligus menjadi dasar untuk proses lanjutan seperti segmentasi baris, pengenalan karakter, dan analisis isi dokumen. Karena itu, pada manuskrip kuno, pengolahan citra digital tidak cukup dipahami sebagai upaya “memperjelas gambar”, tetapi sebagai proses menjaga agar informasi teks tetap dapat dipisahkan, dibaca, dan dianalisis secara konsisten. [1].

B. Binarisasi Klasik

Binarisasi klasik merupakan pendekatan awal yang banyak digunakan untuk memisahkan area *foreground* dan *background* pada citra dokumen. Salah satu metode yang umum digunakan adalah Otsu, yaitu metode *thresholding* global yang menentukan nilai ambang berdasarkan distribusi intensitas piksel pada citra [2]. Metode ini bekerja dengan membagi piksel ke dalam dua kelas, yaitu *foreground* dan *background*, sehingga cocok digunakan ketika perbedaan intensitas antara teks dan latar cukup jelas. Namun, karena hanya menggunakan

satu nilai ambang global, kinerja Otsu dapat menurun ketika dokumen memiliki pencahayaan tidak merata, noda, tekstur latar yang kompleks, atau degradasi lokal.

Selain Otsu, metode Sauvola juga sering digunakan dalam binarisasi dokumen. Berbeda dengan Otsu yang menggunakan satu ambang global, Sauvola menentukan ambang secara lokal berdasarkan karakteristik piksel pada jendela tertentu [3]. Pendekatan ini membuat Sauvola lebih adaptif terhadap variasi latar belakang, terutama pada dokumen yang memiliki tekstur atau pencahayaan tidak seragam. Meski demikian, metode klasik tetap memiliki keterbatasan pada dokumen historis yang mengalami degradasi berat, karena pola kerusakan seperti noda, tinta pudar, dan kontras rendah dapat membuat pemisahan teks dan latar menjadi tidak stabil [12].

C. U-Net

U-Net diperkenalkan pada penelitian [4] sebagai arsitektur *encoder-decoder* yang menggabungkan jalur kontraksi untuk menangkap konteks dan jalur ekspansi untuk memulihkan lokasi objek secara presisi. Lebih detailnya *decoder* atau *expanding path* melakukan rekonstruksi citra setelah *encoder* atau *contracting path* mengambil informasi melalui konvolusi. Ciri yang paling menonjol ada pada *skip connection*, yaitu penghubung antara fitur resolusi tinggi di *encoder* dan *decoder*. Mekanisme ini membuat informasi spasial halus tidak terlalu banyak hilang saat proses *downsampling* berlangsung. Dalam binarisasi dokumen, karakter seperti ini sangat berguna karena model tidak hanya perlu mengenali area teks, tetapi juga menjaga bentuk stroke tipis, tepi aksara, dan lengkungan karakter agar tetap utuh.

D. Resnet

ResNet berangkat dari persoalan yang cukup mendasar dalam *deep learning*, yaitu semakin dalam sebuah jaringan atau *layer*, semakin sulit pula proses optimisasinya. Referensi [6] mengusulkan *residual learning* agar setiap blok jaringan mempelajari residu terhadap *input*-nya, bukan memaksakan diri mempelajari pemetaan penuh dari nol. Pendekatan ini membantu menjaga aliran gradien tetap stabil dan membuat jaringan yang lebih dalam menjadi lebih mudah dilatih. Varian ResNet tersedia dalam beberapa kedalaman, seperti 34, 50, 101, dan 152 layer, dengan *trade-off* antara kapasitas representasi dan biaya komputasi.

E. U-net Resnet 34

U-Net ResNet 34 pada dasarnya memadukan dua kekuatan yang berbeda fungsi. Dari U-Net, model ini mewarisi struktur *decoder* dan *skip connection* yang efektif untuk lokalisasi piksel. Dari ResNet34, model ini memperoleh *encoder residual* yang lebih dalam untuk ekstraksi fitur. Sementara *skip connection* U-Net meneruskan detail resolusi tinggi ke *decoder* agar proses rekonstruksi tidak kehilangan bentuk lokal objek. Dengan susunan seperti itu, model dapat memadukan konteks semantik dari lapisan yang lebih dalam dengan detail spasial dari lapisan yang lebih dangkal. Itu sebabnya arsitektur ini

cocok untuk segmentasi piksel demi piksel, termasuk pada tugas binarisasi citra dokumen [1].

III. METODOLOGI PENELITIAN

A. Pengumpulan Data

Penelitian ini menggunakan dua dataset manuskrip daun lontar yang telah dilengkapi *ground truth* untuk tugas binarisasi citra. Dataset pertama adalah AMADI Lontarset [10], yang telah banyak digunakan pada penelitian analisis manuskrip lontar. Dataset kedua adalah THPLMD (Tamil Handwritten Palm Leaf Manuscript Dataset) sebagaimana dilaporkan pada penelitian [11]. Total data yang digunakan dalam penelitian ini berjumlah 261 citra, yang masing-masing telah memiliki pasangan *ground truth*. Seluruh data kemudian dibagi ke dalam tiga subset, yaitu data latih, data validasi, dan data uji dengan rasio 80:10:10. Data uji dipertahankan sebagai data *holdout* akhir untuk evaluasi performa model, sedangkan data latih dan validasi digunakan dalam proses pelatihan, pemilihan model, serta pengujian tambahan menggunakan 3-fold *cross-validation*.

B. Pre-processing dan Augmentasi Data

Tahap pre-processing pada penelitian ini mengacu pada alur yang digunakan pada penelitian [1]. Setiap citra asli dan *ground truth* terlebih dahulu diberi *padding* agar ukurannya dapat dibagi secara utuh ke dalam blok tanpa *overlap*. Setelah itu, citra dan *ground truth* dipotong menjadi patch berukuran 256×256 piksel. Pendekatan ini dipilih agar seluruh sampel yang masuk ke jaringan memiliki dimensi yang seragam serta lebih efisien untuk proses pelatihan. Pada tahap inferensi, hasil prediksi dari setiap *patch* disusun kembali menggunakan proses *block stitching* sehingga menghasilkan citra binarisasi akhir dengan ukuran yang sesuai dengan citra halaman aslinya.

Untuk meningkatkan variasi data pelatihan, penelitian ini menambahkan data *augmentation* pada data latih. Augmentasi dilakukan secara sinkron antara citra dan *ground truth* agar posisi *foreground* teks tetap sesuai dengan mask acuan. Transformasi yang digunakan meliputi *flip*, rotasi terbatas, penyesuaian *brightness* dan *contrast*, penambahan *noise* ringan, serta blur ringan. Augmentasi hanya diterapkan pada data latih, sedangkan data validasi dan data uji tetap dipertahankan dalam kondisi asli agar proses evaluasi tidak bias.

C. Pengembangan Model

Penelitian ini mengembangkan dan membandingkan lima metode, yaitu *Otsu*, *Sauvola*, *U-Net* dasar, *U-Net ResNet34* tanpa *pretrained*, dan *U-Net ResNet34 pretrained*. *Otsu* dan *Sauvola* digunakan sebagai baseline klasik karena keduanya merupakan metode *thresholding* yang umum digunakan dalam binarisasi dokumen. Kedua metode tersebut tidak melalui proses pelatihan, melainkan langsung diterapkan pada citra uji untuk menghasilkan citra biner. *U-Net* dasar digunakan sebagai baseline utama karena arsitektur ini telah dikenal efektif untuk segmentasi dan binarisasi berbasis piksel. Setiap input berupa

citra RGB berukuran $256 \times 256 \times 3$ dan menghasilkan keluaran berupa *binary mask* $256 \times 256 \times 1$ dengan aktivasi akhir sigmoid.

Model kedua dan ketiga menggunakan arsitektur *U-Net* dengan *encoder ResNet34*. Mengacu pada [1], *encoder ResNet34* terdiri atas lapisan awal konvolusi, *max-pooling*, dan empat tahap *encoder residual*, sedangkan bagian *decoder* tetap mengikuti mekanisme rekonstruksi *U-Net* melalui *upsampling* dan *skip connection*. Pada penelitian ini, dua variasi *encoder residual* diuji, yaitu *ResNet34* tanpa *pretrained* untuk melihat pengaruh arsitektur residual itu sendiri, dan *ResNet34 pretrained* untuk melihat tambahan manfaat dari bobot awal hasil pelatihan sebelumnya. Konfigurasi dari setiap model sama, yakni input $256 \times 256 \times 3$, keluaran $256 \times 256 \times 1$, *batch size* 4, *epoch* 50, aktivasi *sigmoid*, *learning rate* 0.001, dan *Adam* sebagai *optimizer*. Konfigurasi dasar tersebut dipertahankan agar perbandingan tetap setara.

D. Evaluasi

Evaluasi dilakukan dengan membandingkan hasil binarisasi tiap model terhadap *ground truth* pada data uji menggunakan dua aspek utama, yaitu kualitas hasil binarisasi dan efisiensi komputasi. Untuk kualitas binarisasi, metrik yang digunakan meliputi *IoU*, *Dice Coefficient*, *Accuracy*, *Recall*, *Structural Similarity Index Measure* (SSIM), *Feature Similarity Index Measure* (FSIM), *Multi-Scale SSIM* (MS-SSIM), *Gradient Magnitude Similarity Deviation* (GMSD), *Root Mean Square Error* (RMSE), dan *Peak Signal-to-Noise Ratio* (PSNR). *IoU*, *Dice*, *Accuracy*, dan *Recall* digunakan untuk menilai ketepatan klasifikasi piksel *foreground* dan *background*. SSIM, FSIM, MS-SSIM, dan GMSD digunakan untuk menilai kesamaan struktur visual hasil binarisasi terhadap *ground truth*. RMSE dan PSNR digunakan untuk mengukur kesalahan piksel dan kualitas rekonstruksi citra. Selain evaluasi pada data uji, penelitian ini juga menggunakan 3-fold *cross-validation* untuk melihat konsistensi performa model *deep learning* pada pembagian data yang berbeda. *Cross-validation* dilakukan berbasis citra agar *patch* dari citra yang sama tidak tersebar pada fold yang berbeda. Dengan cara ini, hasil validasi menjadi lebih representatif dan dapat mengurangi risiko bias akibat satu pembagian data tertentu.

Untuk memperkuat analisis kuantitatif, penelitian ini juga melakukan uji statistik terhadap hasil metrik per citra. *Friedman test* digunakan untuk membandingkan lebih dari dua metode secara keseluruhan, sedangkan *Wilcoxon signed-rank test* digunakan untuk perbandingan berpasangan antar metode. Uji statistik dilakukan dengan tingkat signifikansi 0,05. Selain itu, penelitian ini juga mengukur efisiensi komputasi melalui waktu pelatihan per *epoch*, total waktu pelatihan, waktu validasi, waktu inferensi pada data uji, jumlah parameter model, serta penggunaan memori komputasi. Seluruh proses pelatihan dan pengujian dilakukan pada perangkat yang sama, yaitu komputer dengan RAM 16 GB, prosesor AMD Ryzen 5 5600, dan GPU NVIDIA GeForce RTX 4060.

IV. HASIL DAN PEMBAHASAN

A. Hasil Cross-Validation dan Uji Statistik

Sebelum evaluasi pada data uji, model deep learning terlebih dahulu dianalisis melalui *3-fold cross-validation* untuk melihat konsistensi proses pelatihan pada pembagian data yang berbeda. *Cross-validation* dilakukan pada tiga model, yaitu *U-Net Standar*, *U-Net ResNet34 tanpa pretrained*, dan *U-Net ResNet34 pretrained ImageNet*. Setiap model dilatih selama 50 epoch pada setiap *fold* dengan skenario pelatihan yang sama. Pada *fold* pertama, data pelatihan terdiri atas 146 citra dengan 7.286 *patch*, sedangkan data validasi terdiri atas 75 citra dengan 3.727 *patch*. Pada *fold* kedua, data pelatihan terdiri atas 146 citra dengan 7.402 *patch* dan data validasi 76 citra dengan 3.611 *patch*. Pada *fold* ketiga, data pelatihan terdiri atas 145 citra dengan 7.338 *patch* dan data validasi 77 citra dengan 3.675 *patch*.

Secara umum, kedua model berbasis *ResNet34* menunjukkan performa validasi yang lebih baik dan lebih stabil dibandingkan *U-Net Standar*. *U-Net ResNet34 pretrained* memperoleh rata-rata *best validation IoU* tertinggi sebesar 0,7054 dengan standar deviasi 0,0179. *U-Net ResNet34 tanpa pretrained* berada sedikit di bawahnya dengan rata-rata *best validation IoU* sebesar 0,6964 dan standar deviasi 0,0122. Sementara itu, *U-Net Standar* memperoleh rata-rata *best validation IoU* sebesar 0,5469 dengan standar deviasi 0,1829. Nilai standar deviasi yang besar pada *U-Net Standar* menunjukkan bahwa performanya kurang stabil antar *fold*. Hal ini terlihat terutama pada *fold* kedua, ketika *U-Net Standar* hanya memperoleh *best validation IoU* sebesar 0,3358 dan *best validation loss* sebesar 5,3879. Sehingga *fold* terbaik untuk setiap model berada pada *fold* 1. Hasil *cross validation* model deep learning ditunjukkan pada Tabel 1.

TABEL I
HASIL CROSS-VALIDATION MODEL DEEP LEARNING

Parameter	U-Net	U-Net ResNet34	U-Net ResNet34 Pretrained
Fold 1 IoU	0,6602	0,7091	0,7246
Fold 2 IoU	0,3358	0,6954	0,7025
Fold 3 IoU	0,6445	0,6847	0,6892
Rata-rata IoU	0,5469	0,6964	0,7054
Std IoU	0,1829	0,0122	0,0179
Rata-rata Val Loss	1,8351	0,0527	0,0528
Rata-rata Best Epoch	26,33	44,00	41,00

Untuk mengetahui apakah perbedaan performa validasi antar model memiliki makna secara statistik, dilakukan *Friedman test* terhadap *best validation IoU* dan *best validation loss*. Pada metrik *best validation IoU*, *Friedman test* menghasilkan nilai statistik 6,0000 dengan p-value 0,0498. Karena nilai p-value lebih kecil dari 0,05, maka terdapat perbedaan performa validasi yang signifikan antar model deep learning. Model

dengan rata-rata terbaik pada metrik ini adalah *U-Net ResNet34 pretrained*. Namun, pada metrik *best validation loss*, *Friedman test* menghasilkan p-value 0,0970, sehingga perbedaan *loss* antar model belum dapat dinyatakan signifikan pada tingkat signifikansi 0,05.

TABEL II
HASIL UJI STATISTIK CROSS-VALIDATION

Metrik	Statistik Friedman	p-value	Keterangan
Best Val IoU	6,0000	0,0498	Signifikan
Best Val Loss	4,6667	0,0970	Tidak signifikan

Berdasarkan Tabel 3, hasil *Wilcoxon* berpasangan menunjukkan bahwa seluruh perbandingan antar model memiliki p-value lebih besar dari 0,05. Hal ini menunjukkan bahwa perbedaan antar pasangan model belum dapat dinyatakan signifikan secara statistik. Meskipun demikian, nilai rata-rata *best validation IoU* tetap menunjukkan kecenderungan bahwa *U-Net ResNet34 pretrained* dan *U-Net ResNet34 tanpa pretrained* memiliki performa validasi lebih baik dibandingkan *U-Net Standar*. Oleh karena itu, hasil *Wilcoxon* digunakan untuk melengkapi analisis berpasangan, sedangkan hasil *Friedman* menunjukkan bahwa secara keseluruhan terdapat perbedaan performa antar model pada metrik *best validation IoU*.

TABEL III
HASIL UJI WILCOXON BERPASANGAN

Metrik	Perbandingan Model	Selisih Rata-rata	p-value
Best Val IoU	Pretrained - ResNet34	0,0090	0,2500
Best Val IoU	Pretrained - U-Net	0,1586	0,2500
Best Val IoU	ResNet34 - U-Net	0,1496	0,2500
Best Val Loss	Pretrained - ResNet34	0,0000	1,0000
Best Val Loss	Pretrained - U-Net	-1,7824	0,2500
Best Val Loss	ResNet34 - U-Net	-1,7824	0,2500

B. Hasil Evaluasi Kinerja Model

Pengujian dilakukan terhadap lima metode, yaitu *Otsu*, *Sauvola*, *U-Net Standar*, *U-Net ResNet34 tanpa pretrained*, dan *U-Net ResNet34 pretrained ImageNet*. *Otsu* dan *Sauvola* digunakan sebagai baseline klasik, sedangkan tiga metode lainnya merupakan model deep learning. Hasil pengujian menunjukkan bahwa metode deep learning, khususnya *U-Net ResNet34 pretrained*, memberikan hasil paling baik pada sebagian besar metrik evaluasi. Sementara itu, baseline klasik masih menunjukkan keterbatasan, terutama pada dokumen dengan degradasi dan variasi latar belakang yang kompleks.

Pada *baseline* klasik, *Otsu* memperoleh *Recall* sangat tinggi sebesar 0,9983, tetapi nilai *IoU*, *Dice*, *Accuracy*, *SSIM*, *RMSE*, dan *PSNR* jauh lebih rendah dibandingkan metode lainnya. Hal ini menunjukkan bahwa *Otsu* cenderung menangkap hampir seluruh piksel teks, tetapi juga menghasilkan banyak kesalahan pada area *background*. *Sauvola* memberikan hasil yang lebih seimbang dibandingkan *Otsu*, dengan *IoU* sebesar 0,3432, *Dice* 0,5018, *Accuracy* 0,9094, *RMSE* 0,2988, dan *PSNR* 10,5591. Meskipun demikian, performa *Sauvola* masih berada di bawah model *deep learning* pada hampir seluruh metrik. Hasil lengkap evaluasi *baseline* klasik ditunjukkan pada Tabel 4.

Pada kelompok model *deep learning*, *U-Net ResNet34* pretrained memperoleh hasil terbaik pada sebagian besar metrik, yaitu *IoU* sebesar 0,7306, *Dice* 0,8263, *Accuracy* 0,9781, *SSIM* 0,9194, *FSIM* 0,9386, *MS-SSIM* 0,9302, *GMSD* 0,1582, *RMSE* 0,1383, dan *PSNR* 17,6995. *U-Net ResNet34* tanpa pretrained berada sedikit di bawahnya dengan *IoU* 0,7238 dan *Dice* 0,8223, tetapi memiliki *Recall* yang hampir sama, yaitu 0,8208. Hasil pada Tabel 5 menunjukkan bahwa penggunaan *encoder residual ResNet34* mampu meningkatkan performa binarisasi dibandingkan *U-Net Standar*, sedangkan pretrained *ImageNet* memberikan tambahan peningkatan terutama pada kualitas struktur visual dan error piksel.

TABEL IV
HASIL EVALUASI MATRIK DEEP LEARNING

Matriks	Model		
	<i>U-Net</i>	<i>ResNet34</i>	<i>ResNet34 pretrained</i>
<i>IoU</i>	0,4623	0,7238	0,7306
<i>Dice</i>	0,5348	0,8223	0,8263
<i>Recall</i>	0,5154	0,8208	0,8212
<i>Accuracy</i>	0,9605	0,9774	0,9781
<i>SSIM</i>	0,8677	0,9171	0,9194
<i>FSIM</i>	0,8356	0,9347	0,9386
<i>MS-SSIM</i>	0,7795	0,9262	0,9302
<i>GMSD</i>	0,2417	0,1621	0,1582
<i>RMSE</i>	0,1863	0,1411	0,1383
<i>PSNR</i>	15,2423	17,5068	17,6995

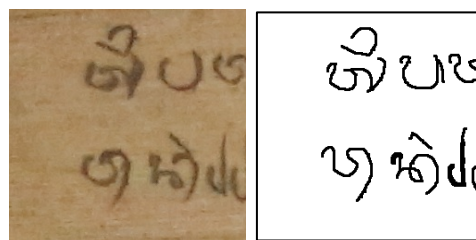
TABEL V
HASIL EVALUASI MATRIK BASELINE KLASIK

Matriks	Model	
	<i>Otsu</i>	<i>Sauvola</i>
<i>IoU</i>	0,1323	0,3432
<i>Dice</i>	0,2309	0,5018
<i>Recall</i>	0,9983	0,6860
<i>Accuracy</i>	0,4955	0,9094
<i>SSIM</i>	0,4080	0,7962
<i>FSIM</i>	0,7231	0,7444
<i>MS-SSIM</i>	0,3554	0,6821
<i>GMSD</i>	0,3955	0,2797
<i>RMSE</i>	0,7000	0,2988
<i>PSNR</i>	3,2111	10,5591

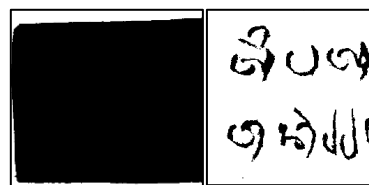
C. Analisis Hasil Visual Binarisasi

Selain evaluasi numerik, hasil binarisasi juga diamati secara visual untuk melihat kemampuan setiap metode dalam

mempertahankan bentuk aksara, memisahkan *foreground* dari *background*, serta mengurangi gangguan latar pada manuskrip historis. Sebagai pembandingan, Gbr. 1 menunjukkan citra asli manuskrip dan *ground truth*. Pada metode klasik, hasil *Otsu* pada Gbr. 2 menunjukkan bahwa hasil binarisasi masih didominasi oleh satu kelas sehingga pemisahan antara teks dan latar belum terlihat jelas., tetapi masih membawa banyak bagian latar sebagai *foreground* sehingga hasil binarisasi tampak kurang bersih. *Sauvola* menghasilkan pemisahan yang lebih seimbang dibandingkan *Otsu* karena sebagian *noise* latar dapat berkurang, tetapi beberapa bagian *stroke* masih belum stabil dan hasilnya belum sepenuhnya mendekati *ground truth*.

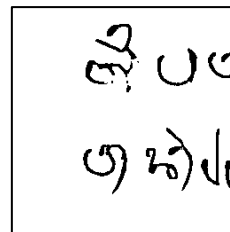


Gbr. 1. Gambar citra asli dan *ground truth*

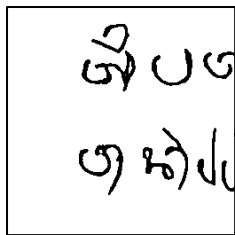


Gbr. 2. Gambar hasil dari *Otsu* (kiri) dan *Sauvola* (kanan)

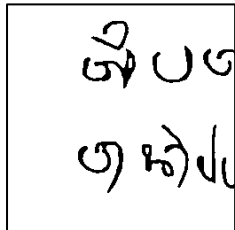
Pada model *deep learning*, *U-Net Standar* pada Gbr. 3 menghasilkan latar yang lebih bersih dibandingkan metode klasik, tetapi beberapa bagian aksara yang tipis masih tampak kurang utuh. *U-Net ResNet34* tanpa pretrained pada Gbr. 4 menunjukkan hasil yang lebih baik, dengan bentuk aksara yang lebih utuh dan *noise* latar yang lebih terkendali. Sementara itu, *U-Net ResNet34 pretrained* pada Gbr. 7 memberikan hasil visual yang paling stabil, terutama pada kerapian tepi aksara, kebersihan latar, dan kemiripan bentuk terhadap *ground truth*. Secara keseluruhan, hasil visual memperlihatkan bahwa metode klasik masih terbatas dalam menghadapi degradasi dan variasi latar, sedangkan model berbasis *encoder residual* mampu menghasilkan binarisasi yang lebih konsisten dan lebih dekat dengan mask acuan.



Gbr. 3. Gambar hasil dari *U-net* dasar



Gbr. 4. Gambar hasil dari ResNet34 tanpa pretrained



Gbr. 5. Gambar hasil dari ResNet34 pretrained

D. Efisiensi Komputasi

Dari sisi kompleksitas model, *Otsu* dan *Sauvola* tidak memiliki parameter pelatihan karena keduanya merupakan metode *thresholding* klasik. Pada model *deep learning*, *U-Net Standar* memiliki jumlah parameter terbesar, yaitu 31.031.745 parameter dengan ukuran model 118,38 MB. Sebaliknya, *U-Net ResNet34* tanpa *pretrained* dan *U-Net ResNet34 pretrained* memiliki jumlah parameter yang sama, yaitu 24.436.369 parameter dengan ukuran model 93,29 MB. Hal ini menunjukkan bahwa pada implementasi penelitian ini, penggunaan *encoder ResNet34* tidak hanya meningkatkan performa, tetapi juga menghasilkan model yang lebih ringan dibandingkan *U-Net Standar*.

Dari sisi waktu pelatihan, *U-Net Standar* membutuhkan rata-rata total waktu pelatihan paling besar, yaitu 16.484,91 detik per *fold*. *U-Net ResNet34* tanpa *pretrained* membutuhkan 6.469,98 detik, sedangkan *U-Net ResNet34 pretrained* membutuhkan 6.083,90 detik. Dengan demikian, kedua varian *U-Net ResNet34* lebih efisien daripada *U-Net Standar* dalam proses pelatihan. Detail hasil ditulis pada Tabel 6.

TABEL VI
KOMPLEKSITAS MODEL DAN WAKTU PELATIHAN

Parameter	<i>U-Net</i>	<i>U-Net ResNet34</i>	<i>U-Net ResNet34 (pretrained)</i>
Total Parameter	31.031.745	24.436.369	24.436.369
Trainable Parameter	31.031.745	24.436.369	24.436.369
Ukuran Model (MB)	118,38	93,29	93,29
Best Epoch	26,33	44,00	41,00
Total Training Time (detik)	16.484,91	6.469,98	6.083,90
Rata-rata Waktu Epoch (detik)	329,70	129,40	121,68

Pada tahap inferensi, metode *Otsu* menjadi metode tercepat karena hanya membutuhkan 0,31 detik untuk memproses 27 citra uji. *Sauvola* membutuhkan waktu lebih besar, yaitu 4,52 detik, karena proses *thresholding* dilakukan secara lokal. Pada model *deep learning*, *U-Net ResNet34* tanpa *pretrained* menjadi model tercepat dengan rata-rata waktu inferensi 6,08 detik, diikuti *U-Net ResNet34 pretrained* sebesar 8,85 detik, dan *U-Net Standar* sebesar 17,38 detik. Kecepatan ini juga terlihat dari nilai *patches per second*, di mana *U-Net ResNet34* tanpa *pretrained* mencapai 209,01 *patches/detik*, lebih tinggi dibandingkan *U-Net ResNet34 pretrained* sebesar 168,51 *patches/detik* dan *U-Net Standar* sebesar 73,15 *patches/detik*. Dari sisi penggunaan memori GPU, *U-Net Standar* memiliki penggunaan memori paling besar, baik saat pelatihan maupun inferensi. *Peak GPU train* pada *U-Net Standar* mencapai 2.685,11 MB, sedangkan dua varian *U-Net ResNet34* berada pada 1.353,85 MB. Pada inferensi, *U-Net ResNet34* tanpa *pretrained* menggunakan memori paling rendah, yaitu 194,02 MB, sedangkan *U-Net ResNet34 pretrained* sebesar 904,43 MB dan *U-Net Standar* sebesar 1.854,06 MB. Hasil ini menunjukkan bahwa *U-Net ResNet34* tanpa *pretrained* merupakan pilihan yang paling efisien dari sisi kecepatan inferensi dan penggunaan memori, sedangkan *U-Net ResNet34 pretrained* memberikan kualitas metrik terbaik. Detail efisiensi dan penggunaan memori ditunjukkan pada Tabel 7.

TABEL VII
EFISIENSI INFERENSI DAN PENGGUNAAN MEMORI

Parameter	<i>U-Net</i>	<i>U-Net ResNet34</i>	<i>U-Net ResNet34 (pretrained)</i>
Waktu Inferensi (detik)	17,38	6,08	8,85
Patches/detik	73,15	209,01	168,51
Images/detik	1,56	4,44	3,58
ms/patch	13,68	4,78	6,97
ms/image	643,67	225,05	327,70
Peak GPU Train (MB)	2.685,11	1.353,85	1.353,85
Peak GPU Inferensi (MB)	1.854,06	194,02	904,43

V. KESIMPULAN

Berdasarkan pengujian, model *deep learning* memberikan hasil binarisasi yang lebih baik dibandingkan metode klasik *Otsu* dan *Sauvola*. Metode *Otsu* masih kurang mampu memisahkan teks dan latar secara jelas, sedangkan *Sauvola* menghasilkan pemisahan yang lebih seimbang tetapi tetap belum mampu menyamai model *deep learning*. Pada kelompok model *deep learning*, *U-Net ResNet34 pretrained* memperoleh hasil terbaik pada sebagian besar metrik evaluasi, termasuk IoU, *Dice*, *Accuracy*, SSIM, FSIM, MS-SSIM, GMSD, RMSE, dan PSNR. *U-Net ResNet34* tanpa *pretrained* juga menunjukkan performa yang sangat dekat dan lebih baik daripada *U-Net Standar*. Hasil *cross-validation* memperlihatkan bahwa kedua model berbasis *ResNet34* memiliki performa validasi yang

lebih stabil dibandingkan *U-Net* Standar. Uji *Friedman* juga menunjukkan adanya perbedaan signifikan antar model *deep learning* pada metrik *best validation* IoU, meskipun uji *Wilcoxon* berpasangan belum menunjukkan perbedaan signifikan.

Dari sisi efisiensi komputasi, metode klasik tetap menjadi yang paling ringan karena tidak memiliki proses pelatihan dan parameter model. Namun, dari sisi kualitas hasil binarisasi, model *deep learning* jauh lebih unggul. Di antara model *deep learning*, *U-Net* Standar menjadi model paling berat dan paling lambat pada implementasi ini, sedangkan kedua varian *U-Net ResNet34* memiliki jumlah parameter lebih kecil, waktu pelatihan lebih cepat, serta inferensi yang lebih efisien. *U-Net ResNet34* tanpa *pretrained* menjadi pilihan yang seimbang ketika prioritas utama adalah efisiensi inferensi dan penggunaan memori, sedangkan *U-Net ResNet34 pretrained* menjadi pilihan terbaik ketika prioritas utama adalah kualitas hasil binarisasi. Dengan demikian, penelitian ini menunjukkan bahwa *encoder residual* berperan penting dalam meningkatkan performa *U-Net* pada binarisasi manuskrip historis, sementara penggunaan *pretrained* memberikan tambahan peningkatan terutama pada kualitas hasil akhir.

REFERENSI

- [1] I. Yuadi, K. Nisa', N. U. Nazikhah, Y. A. Halim, A. T. Asyhari, and C.-C. Hu, "A Benchmark Study of Classical and U-Net ResNet34 Methods for Binarization of Balinese Palm Leaf Manuscripts," *Heritage*, vol. 8, art. no. 337, 2025, doi: 10.3390/heritage8080337.
- [2] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979, doi: 10.1109/TSMC.1979.4310076.
- [3] J. J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern Recognition*, vol. 33, no. 2, pp. 225–236, 2000, doi: 10.1016/S0031-3203(99)00055-2.
- [4] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*, N. Navab et al., Eds., LNCS, vol. 9351. Cham, Switzerland: Springer, 2015, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.
- [5] P. V. Bezmaternykh, D. A. Il'in, and D. P. Nikolaev, "U-Net-bin: hacking the document image binarization contest," *Computer Optics*, vol. 43, no. 5, pp. 825–832, 2019, doi: 10.18287/2412-6179-2019-43-5-825-832.
- [6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
- [7] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson, 2018.
- [8] A. N. L. Hanum, A. T. Priyadi, A. N. Hanum, and A. A. Akbar, "Peran library, archives, museums dalam pelestarian naskah kuno di Kalimantan Barat," *Berkala Ilmu Perpustakaan dan Informasi*, vol. 18, no. 2, 2022, doi: 10.22146/bip.v18i2.6294.
- [9] O. Rizkiantha, R. Rusmiatiningsih, C. Afrina, and M. Oktapia, "Preservasi Naskah Kuno: Strategi Pelestarian Minimal untuk Mempertahankan Warisan Budaya," *Jurnal Pustaka Budaya*, vol. 11, no. 1, 2024.
- [10] M. Windu et al., "AMADI LontarSet: The First Handwritten Balinese Palm Leaf Manuscripts Dataset," in *Proc. 15th Int. Conf. Frontiers in Handwriting Recognition (ICFHR)*, 2016, pp. 168–173, doi: 10.1109/ICFHR.2016.39.
- [11] I. Jailingeswari and S. Gopinathan, "Tamil handwritten palm leaf manuscript dataset (THPLMD)," *Data Brief*, vol. 53, Art. no. 110100, 2024, doi: 10.1016/j.dib.2024.110100.
- [12] A. Sulaiman, K. Omar, and M. F. Nasrudin, "Degraded Historical Document Binarization: A Review on Issues, Challenges, Techniques, and Future Directions," *J. Imaging*, vol. 5, no. 4, Art. no. 48, 2019, doi: 10.3390/jimaging5040048.
- [13] M. Almeida, R. D. Lins, R. Bernardino, D. Jesus, and B. Lima, "A New Binarization Algorithm for Historical Documents," *J. Imaging*, vol. 4, no. 2, Art. no. 27, 2018, doi: 10.3390/jimaging4020027.
- [14] C. Tensmeyer and T. R. Martinez, "Document Image Binarization with Fully Convolutional Neural Networks," in *Proc. 14th IAPR Int. Conf. Document Analysis and Recognition (ICDAR)*, 2017, pp. 99–104, doi: 10.1109/ICDAR.2017.25.
- [15] J. Calvo-Zaragoza and A.-J. Gallego, "A selectional auto-encoder approach for document image binarization," *Pattern Recognit.*, vol. 86, pp. 37–47, 2019, doi: 10.1016/j.patcog.2018.08.011.
- [16] F. Damayanti, E. M. Yuniarno, and Y. K. Suprpto, "Modified ResUNet Architecture for Binarization in Degraded Javanese Ancient Manuscript," *Math. Model. Eng. Probl.*, vol. 11, no. 7, pp. 1758–1772, 2024, doi: 10.18280/mmep.110706.
- [17] S. Han, S. Ji, and J. Rhee, "Diffusion-Denoising Process with Gated U-Net for High-Quality Document Binarization," *Appl. Sci.*, vol. 13, no. 20, Art. no. 11141, 2023, doi: 10.3390/app132011141.
- [18] L. Zhang, K. Wang, and Y. Wan, "An Efficient Transformer-CNN Network for Document Image Binarization," *Electronics*, vol. 13, no. 12, Art. no. 2243, 2024, doi: 10.3390/electronics13122243.