

UTS
PENGOLAHAN CITRA



NAMA : Michael Christia Putra

NIM : 202331203

KELAS : A

DOSEN : Dr. Dra. Dwina Kuswardani, M.Kom

NO.PC : 18

ASISTEN : 1. Clarenca Sweetdiva Pereira

2. Viana Salsabila Fairuz Syahla

3. Kashrina Masyid Azka

4. Sasikirana Ramadhanty Setiawan Putri

INSTITUT TEKNOLOGI PLN
TEKNIK INFORMATIKA
2024/2025

Isi

BAB I	3
PENDAHULUAN	3
1.1 Rumusan Masalah	3
1.1 Tujuan Masalah	3
1.2 Manfaat Masalah	3
BAB II	4
LANDASAN TEORI	4
II.I. Pengantar Pengolahan Citra Digital	4
II.II. Representasi dan Model Warna dalam Citra Digital	4
II.III. Teknik Splitting Warna dan Penghilangan Channel	5
II.IV. Teknik Peningkatan Kecerahan dan Kontras Citra	6
II.V. Visualisasi Hasil Pengolahan dengan Matplotlib	7
BAB III	8
HASIL	8
III.I. Analisis Tiga Gambar Grayscale dengan Indeks Warna Hilang	8
III.II. Sistem Tresholding pada Citra Binary	9
III.III. Peningkatan Citra Backlight Menjadi Citra Berkualitas Baik	10
BAB IV	12
PENUTUP	12
DAFTAR PUSTAKA	Kesalahan! Bookmark tidak ditentukan.

BAB I

PENDAHULUAN

1.1 Rumusan Masalah

- ☐ Bagaimana representasi warna dalam citra digital dapat dipisahkan menjadi channel RGB?
- ☐ Bagaimana pengaruh penghilangan salah satu channel warna terhadap tampilan akhir citra?
- ☐ Bagaimana proses peningkatan kecerahan dan kontras dapat diterapkan untuk memperbaiki kualitas visual citra?
- ☐ Bagaimana visualisasi hasil pengolahan citra, baik dalam bentuk citra maupun histogram, dapat membantu dalam analisis kualitas citra?

1.1 Tujuan Masalah

- ☐ Menjelaskan cara kerja pemisahan warna RGB dalam citra digital.
- ☐ Menunjukkan dampak penghilangan channel warna terhadap komposisi citra.
- ☐ Menerapkan teknik peningkatan kecerahan dan kontras pada citra digital untuk memperoleh hasil visual yang lebih baik.
- ☐ Menampilkan hasil pengolahan citra dan histogram intensitas untuk keperluan analisis dan interpretasi.

1.2 Manfaat Masalah

Manfaat dari penulisan laporan ini adalah untuk memberikan pemahaman yang lebih mendalam mengenai teknik dasar dalam pengolahan citra digital, khususnya terkait representasi warna, transformasi visual, serta metode visualisasi hasil menggunakan perangkat lunak berbasis Python seperti OpenCV dan Matplotlib. Dengan pemahaman ini, mahasiswa diharapkan mampu melakukan modifikasi dan analisis citra untuk berbagai keperluan teknis dan riset, serta menjadi dasar bagi penerapan yang lebih kompleks seperti segmentasi objek dan pengenalan pola.

BAB II

LANDASAN TEORI

II.I. Pengantar Pengolahan Citra Digital

Pengolahan citra digital merupakan cabang ilmu yang mengkaji cara untuk memanipulasi dan menganalisis gambar yang dihasilkan melalui perangkat digital. Definisi pengolahan citra digital mencakup berbagai teknik yang digunakan untuk memperbaiki kualitas gambar, mengekstraksi informasi, dan melakukan analisis menyeluruh terhadap citra. Ruang lingkupnya mencakup konversi citra analog menjadi digital, pemrosesan gambar dalam bentuk dua dimensi, serta penerapan algoritma untuk meningkatkan kualitas visual dan analitis dari citra tersebut. Sebagaimana yang diungkapkan oleh Batchelor (2022) [1], pengolahan citra digital meliputi teknik-teknik seperti pengurangan noise, peningkatan kontras, dan pengenalan pola, yang semua bertujuan untuk menghasilkan informasi yang lebih jelas dan akurat dari gambar yang diolah.

Perbedaan mendasar antara pengolahan citra analog dan digital terletak pada cara data citra diproses. Pengolahan citra analog bekerja dengan sinyal kontinu, yang rentan terhadap gangguan dan degradasi kualitas, sedangkan pengolahan citra digital menggunakan representasi diskrit yang memungkinkan penerapan berbagai algoritma matematis untuk pemrosesan yang lebih efektif dan efisien. Kumar dan Professor (2023) [2] menjelaskan bahwa dengan digitalisasi, kita dapat menggunakan algoritma yang kompleks untuk meningkatkan kualitas gambar dan melakukan analisis mendalam tanpa khawatir akan akumulasi noise yang sering terjadi pada pengolahan citra analog. Oleh karena itu, pengolahan citra digital menjadi solusi yang lebih unggul dalam banyak aplikasi, termasuk dalam bidang medis, industri, dan keamanan.

Aplikasi pengolahan citra digital sangat beragam dan meluas di berbagai bidang. Dalam dunia medis, misalnya, teknologi ini digunakan untuk menganalisis citra medis seperti MRI dan CT scan, yang memungkinkan dokter untuk mengidentifikasi penyakit dengan lebih akurat dan cepat. Di sektor industri, pengolahan citra digunakan untuk kontrol kualitas produk, di mana citra produk dianalisis untuk mendeteksi cacat secara otomatis. Selain itu, di bidang keamanan, teknologi pengenalan wajah dan pemantauan video menggunakan pengolahan citra digital untuk meningkatkan keamanan publik. Huang (2022) [3] menekankan bahwa kemajuan teknologi dalam pengolahan citra digital tidak hanya meningkatkan efisiensi, tetapi juga membuka peluang baru untuk inovasi dalam berbagai aplikasi. Dengan demikian, pengolahan citra digital tidak hanya menjadi alat yang penting dalam analisis visual, tetapi juga pendorong kemajuan di banyak bidang lainnya.

II.II. Representasi dan Model Warna dalam Citra Digital

Model warna RGB (Red, Green, Blue) merupakan salah satu dasar utama dalam representasi citra digital. Dalam model ini, setiap citra dibangun dari kombinasi tiga warna dasar: merah, hijau, dan biru. Sistem ini berfungsi dengan cara menggabungkan intensitas masing-masing channel untuk menciptakan berbagai warna yang dapat dilihat. RGB adalah model aditif, di mana warna dihasilkan dengan menambahkan cahaya dari ketiga channel tersebut. Setiap channel memiliki rentang nilai yang umumnya berkisar antara 0 hingga 255, yang mewakili intensitas cahaya dari masing-masing warna. Menurut Li dan Zhu (2020) [4], sistem ini sangat efisien dalam merepresentasikan warna karena mata manusia paling sensitif terhadap kombinasi warna-warna ini, memungkinkan kita untuk

melihat spektrum warna yang luas hanya dengan tiga channel. Dalam konteks citra digital, penggunaan RGB sangat luas, mulai dari fotografi digital, pemrosesan video, hingga aplikasi grafis.

Channel warna dalam model RGB masing-masing memberikan kontribusi unik terhadap warna akhir yang terlihat. Channel merah, hijau, dan biru berfungsi secara sinergis; misalnya, kombinasi maksimal dari ketiga channel (255, 255, 255) menghasilkan warna putih, sedangkan ketiga channel dengan nilai nol (0, 0, 0) menghasilkan warna hitam. Perbedaan intensitas antara channel-channel ini memungkinkan pembuatan warna yang kompleks. Namun, meskipun efektif, model RGB memiliki keterbatasan, terutama dalam hal pengolahan citra yang memerlukan analisis lebih dalam mengenai intensitas cahaya. Khudhair et al. (2023) [5] menunjukkan bahwa dalam beberapa aplikasi, konversi ke skala keabuan (grayscale) lebih bermanfaat karena menyederhanakan informasi visual tanpa kehilangan detail penting, terutama dalam analisis bentuk dan tekstur.

Perbedaan antara model RGB dan grayscale sangat signifikan. Model grayscale menyajikan citra dalam satu channel, di mana setiap piksel hanya memiliki satu nilai intensitas yang merepresentasikan kecerahan dari hitam ke putih. Konversi dari RGB ke grayscale biasanya dilakukan dengan menggunakan formula tertentu yang mempertimbangkan sensitivitas mata manusia terhadap warna. Misalnya, formula yang umum digunakan adalah mengalikan setiap channel dengan faktor tertentu: 0.299 untuk merah, 0.587 untuk hijau, dan 0.114 untuk biru, yang mencerminkan sensitivitas visual manusia terhadap ketiga warna tersebut. Hal ini bertujuan untuk mempertahankan detail penting dalam citra saat beralih ke grayscale. Kong et al. (2022) [6] menekankan bahwa konversi ini tidak hanya mengurangi kompleksitas data tetapi juga memungkinkan aplikasi yang lebih efektif dalam analisis citra. Dengan menghilangkan informasi warna, fokus dapat dialihkan ke struktur dan pola yang mungkin lebih relevan dalam konteks pemrosesan citra tertentu, seperti deteksi tepi atau segmentasi objek.

II.III. Teknik Splitting Warna dan Penghilangan Channel

Proses splitting warna, atau pemisahan channel warna, merupakan teknik penting dalam pengolahan citra yang memungkinkan analisis mendalam terhadap setiap komponen warna dalam citra digital. Dalam model RGB, setiap citra terdiri dari tiga channel: merah, hijau, dan biru. Teknik ini memungkinkan kita untuk memisahkan ketiga channel tersebut sehingga setiap channel dapat diolah atau dianalisis secara terpisah. Dengan menggunakan fungsi seperti `cv2.split()` dalam OpenCV, kita dapat dengan mudah membagi citra menjadi tiga array terpisah yang masing-masing mewakili intensitas warna merah, hijau, dan biru. Proses ini sangat berguna ketika kita ingin fokus pada satu komponen warna tertentu dan mengeksplorasi bagaimana komponen tersebut berkontribusi terhadap keseluruhan citra. Sebagai contoh, jika kita ingin meningkatkan kontras warna hijau dalam citra tanaman, kita dapat memisahkan channel hijau dan menerapkan teknik pengolahan citra khusus hanya pada channel tersebut.

Tujuan dari teknik splitting ini tidak hanya untuk analisis warna, tetapi juga untuk mempermudah manipulasi citra berdasarkan channel yang terpisah. Dengan memisahkan channel, kita dapat melakukan berbagai operasi pemrosesan, seperti filter, peningkatan kontras, atau deteksi tepi, yang lebih fokus dan efektif. Misalnya, dalam konteks deteksi objek, kita mungkin tertarik hanya pada channel merah untuk mengidentifikasi objek yang memiliki warna dominan merah. Proses ini membawa kita ke teknik penghilangan channel, di mana kita dapat mengatur nilai salah satu channel menjadi nol. Menghilangkan channel merah, misalnya, berarti kita mengabaikan informasi warna

merah dalam citra. Ini dapat dilakukan dengan sederhana dengan manipulasi array, di mana kita mengatur semua nilai pada channel merah menjadi nol, yang akan menghasilkan citra yang hanya menampilkan informasi dari channel hijau dan biru.

Implikasi visual dari penghilangan salah satu channel warna cukup signifikan. Ketika channel merah dihilangkan, citra yang dihasilkan akan kehilangan semua informasi yang terkait dengan warna merah, yang bisa mengubah secara drastis warna dan tampilan objek dalam citra. Hasilnya adalah citra yang tampak lebih dingin dan mungkin kehilangan detail penting yang sebelumnya terwakili oleh channel merah. Wang et al. (2025) [7] menunjukkan bahwa teknik penghilangan channel bisa digunakan dalam aplikasi watermarking, di mana manipulasi channel bisa membantu dalam menyisipkan informasi tambahan tanpa mengganggu tampilan visual secara berlebihan. Dengan menggunakan teknik ini, kita bisa menguji berbagai efek visual dan analisis yang berbeda, memungkinkan pemahaman yang lebih baik tentang bagaimana setiap channel berkontribusi terhadap citra keseluruhan. Menggunakan OpenCV untuk teknik ini memberikan fleksibilitas dan kemudahan, sehingga para peneliti dan praktisi dapat dengan cepat bereksperimen dengan berbagai manipulasi channel untuk mencapai hasil yang diinginkan dalam pengolahan citra.

II.IV. Teknik Peningkatan Kecerahan dan Kontras Citra

Peningkatan kecerahan dan kontras citra adalah dua teknik penting dalam pengolahan citra digital yang berfungsi untuk meningkatkan visibilitas dan kualitas visual dari gambar. Kecerahan (brightness) mengacu pada seberapa terang atau gelapnya suatu citra secara keseluruhan, sedangkan kontras merujuk pada perbedaan antara piksel terang dan gelap dalam citra tersebut. Kecerahan yang tinggi berarti citra tampak lebih terang, sementara kontras yang tinggi berarti terdapat perbedaan yang mencolok antara bagian terang dan gelap dari citra. Menurut Hassan (2022) [8], pemahaman yang jelas tentang kedua konsep ini sangat penting untuk menerapkan teknik-teknik peningkatan yang tepat, karena keduanya mempengaruhi bagaimana citra ditangkap dan ditampilkan pada perangkat digital. Tanpa peningkatan yang tepat, citra mungkin terlihat buram dan tidak informatif, yang dapat mengurangi pengalaman pengguna dalam melihat dan menganalisis gambar.

Salah satu metode untuk meningkatkan kecerahan citra adalah dengan menambah nilai piksel pada citra tersebut. Proses ini dapat dilakukan dengan menambahkan konstanta ke setiap nilai piksel dalam citra, sehingga meningkatkan kecerahan keseluruhan. Teknik lain yang lebih canggih adalah penggunaan transformasi linear, yang dapat dinyatakan dengan rumus $g(x,y) = \alpha f(x,y) + \beta$. Dalam rumus ini, $g(x,y)$ adalah nilai piksel hasil, $f(x,y)$ adalah nilai piksel awal, α adalah parameter yang mengontrol kontras, dan β adalah parameter yang mengontrol kecerahan. Parameter α (kontras) menentukan seberapa besar perbedaan antara nilai piksel, sementara β (kecerahan) menentukan seberapa terang citra tersebut setelah transformasi. Peningkatan kontras dapat membuat detail dalam citra lebih terlihat, sedangkan peningkatan kecerahan akan memberikan tampilan lebih cerah. Parimala et al. (2024) [9] menjelaskan bahwa dengan mengatur kedua parameter ini, kita dapat mencapai hasil yang optimal dalam meningkatkan kualitas citra, terutama dalam konteks citra yang diambil dalam kondisi pencahayaan yang buruk.

Dalam praktiknya, penggunaan OpenCV untuk menerapkan teknik ini sangat umum. Fungsi seperti `cv2.convertScaleAbs()` memungkinkan pengguna untuk dengan mudah menerapkan transformasi linear pada citra. Fungsi ini menggabungkan pengubahan skala dan penambahan offset sekaligus, sehingga memudahkan dalam meningkatkan kecerahan dan kontras. Manipulasi manual

juga dapat dilakukan dengan mengakses setiap nilai piksel dalam array citra dan menerapkan rumus di atas. Dampak dari peningkatan kecerahan dan kontras terhadap citra buram dapat sangat signifikan; citra yang sebelumnya tampak tidak jelas dan kehilangan detail dapat menjadi lebih informatif dan jelas. Namun, perlu diingat bahwa peningkatan yang berlebihan dapat menyebabkan distorsi pada citra, seperti hilangnya informasi di area yang terlalu terang atau terlalu gelap. Oleh karena itu, penting untuk menemukan keseimbangan yang tepat dalam pengaturan parameter α dan β agar hasil yang diperoleh optimal dan citra tetap terlihat alami.

II.V. Visualisasi Hasil Pengolahan dengan Matplotlib

Matplotlib adalah salah satu pustaka Python yang paling populer untuk visualisasi data, termasuk citra digital. Salah satu fungsi utama dalam Matplotlib adalah `matplotlib.pyplot.imshow()`, yang digunakan untuk menampilkan citra dalam format yang mudah dipahami. Dengan menggunakan fungsi ini, pengguna dapat menampilkan citra dalam berbagai format, baik warna maupun grayscale. Fungsi ini mengubah data citra menjadi representasi visual yang dapat dianalisis dan dievaluasi secara langsung. Sebagaimana dijelaskan oleh Susanto dan Utami (2023) [10], kemampuan Matplotlib untuk menampilkan citra secara efektif sangat penting dalam analisis citra medis, seperti dalam studi radiografi dada untuk mendeteksi kanker paru-paru. Dengan `imshow()`, pengguna dapat dengan cepat melihat hasil pengolahan citra dan memperoleh pemahaman yang lebih baik tentang data yang sedang dianalisis.

Ketika menampilkan citra, terdapat perbedaan penting antara citra warna dan grayscale di Matplotlib. Citra warna biasanya ditampilkan dalam format RGB, di mana setiap piksel memiliki tiga komponen warna yang berbeda—merah, hijau, dan biru. Sebaliknya, citra grayscale hanya memiliki satu komponen intensitas yang merepresentasikan kecerahan dari hitam ke putih. Untuk menampilkan citra grayscale dengan benar, pengguna harus menyesuaikan `colormap` menggunakan parameter `cmap='gray'` dalam fungsi `imshow()`. Hal ini memungkinkan citra grayscale ditampilkan dengan nuansa yang sesuai, sehingga memudahkan interpretasi visual. Tomaszewski dan Dajka (2021) [11] menunjukkan bahwa penyesuaian yang tepat terhadap `colormap` dapat meningkatkan visibilitas detail penting dalam citra histologis, membantu dalam identifikasi perubahan yang signifikan dalam struktur jaringan.

Selain menampilkan citra, Matplotlib juga memungkinkan visualisasi perbandingan hasil pengolahan citra melalui penggunaan subplots dengan `plt.subplot()`. Dengan memanfaatkan subplots, pengguna dapat menampilkan beberapa citra dalam satu tampilan, memberikan konteks yang lebih luas bagi analisis. Misalnya, pengguna dapat membandingkan citra asli dengan citra yang telah diproses untuk menilai efektivitas metode pengolahan yang diterapkan. Selain itu, menampilkan histogram sebagai alat bantu analisis citra juga sangat berguna. Fungsi seperti `cv2.calcHist()` dapat digunakan untuk menghitung histogram dari citra, yang kemudian dapat ditampilkan di subplot yang sama atau terpisah. Histogram memberikan informasi tambahan tentang distribusi intensitas piksel dalam citra, memungkinkan pengguna untuk mengevaluasi perubahan yang terjadi setelah pengolahan. Dengan cara ini, Matplotlib tidak hanya berfungsi sebagai alat visualisasi, tetapi juga sebagai platform yang mendukung analisis data yang lebih mendalam dan informatif.

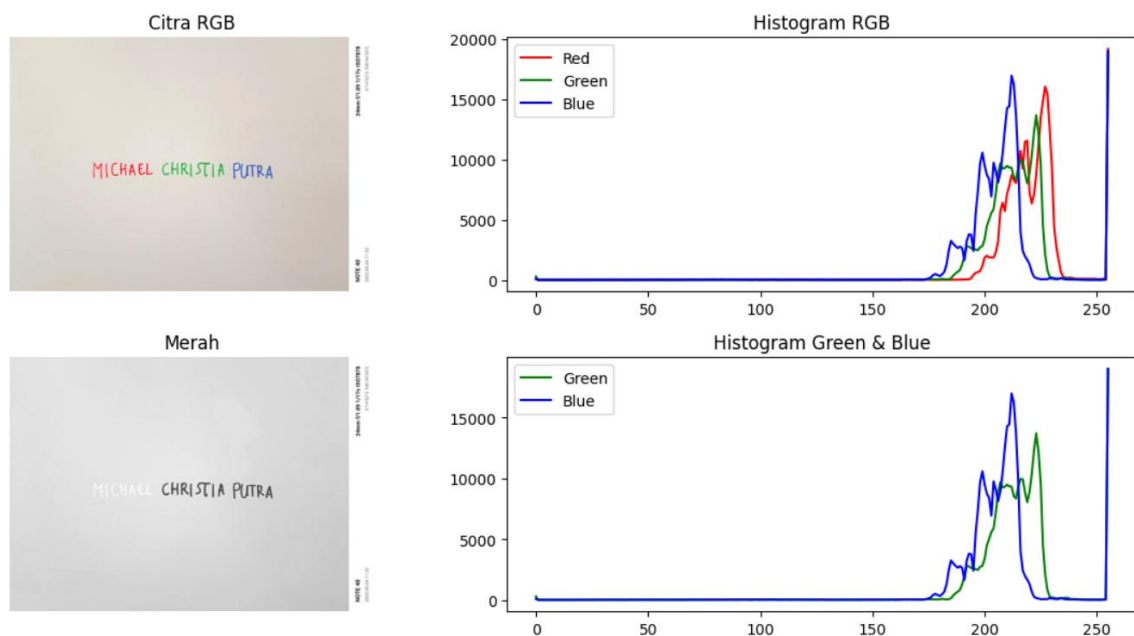
BAB III

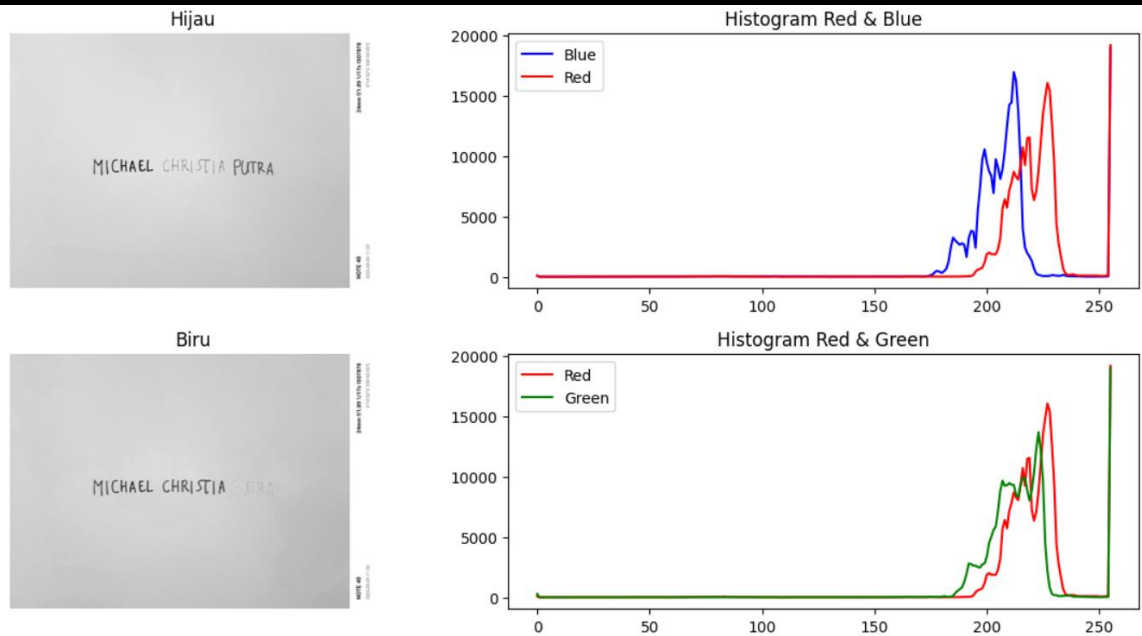
HASIL

III.I. Analisis Tiga Gambar Grayscale dengan Indeks Warna Hilang

Dalam bagian ini, saya menganalisis tiga citra grayscale yang masing-masing kehilangan salah satu indeks warnanya: merah, hijau, dan biru. Gambar pertama menunjukkan hasil di mana indeks merah hilang, dengan nilai piksel yang terdistribusi antara 200 hingga 240. Ketika kami menampilkan histogram untuk gambar ini, terlihat bahwa distribusi intensitas warna merah menunjukkan kekosongan pada rentang yang seharusnya mewakili warna merah, menegaskan bahwa informasi warna tersebut telah hilang. Citra ini tampak lebih dingin dan kehilangan detail penting yang biasanya diwakili oleh warna merah, sehingga memberikan kesan yang berbeda dibandingkan dengan citra asli.

Pada gambar kedua, di mana indeks hijau hilang, histogram menunjukkan nilai intensitas berkisar antara 180 hingga 230. Dalam hal ini, hilangnya indeks hijau menciptakan citra yang cenderung lebih gelap dan kurang hidup. Histogram menunjukkan penurunan signifikan pada nilai intensitas hijau, yang berkontribusi pada persepsi visual citra. Akibatnya, citra ini tampak tidak seimbang, dan detail yang biasanya terlihat jelas dalam warna hijau menjadi tidak terlihat. Gambar ketiga, yang kehilangan indeks biru, menunjukkan histogram dengan rentang intensitas antara 170 hingga 210. Di sini, hilangnya warna biru menciptakan tampilan yang lebih pudar, di mana kontras antara elemen dalam citra menjadi kurang jelas. Ketiga gambar ini menunjukkan bagaimana kehilangan satu channel warna dapat secara signifikan mempengaruhi visualisasi dan analisis citra, serta menyoroti pentingnya setiap channel dalam konteks representasi warna yang utuh.

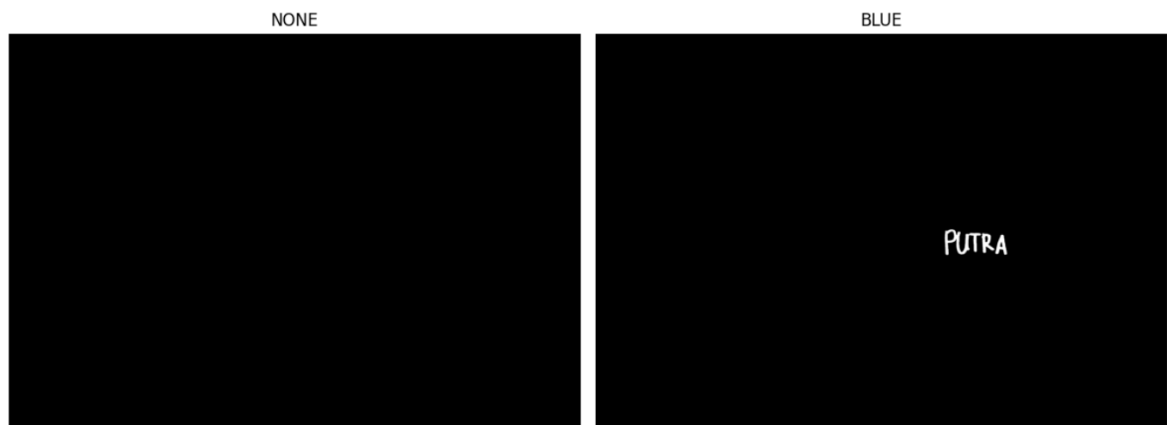




III.II. Sistem Tresholding pada Citra Binary

Pada bagian ini, saya membahas penerapan sistem tresholding pada citra binary. Tresholding merupakan teknik penting dalam pengolahan citra yang digunakan untuk memisahkan objek dari latar belakang berdasarkan nilai ambang tertentu. Dalam praktiknya, semakin tinggi nilai ambang yang diberikan, warna pada citra akan mulai terlihat secara bertahap, dimulai dari warna biru, kemudian merah, dan akhirnya hijau. Proses ini membantu dalam mengidentifikasi dan menyoroti elemen penting dalam citra yang mungkin tersembunyi dalam latar belakang yang kompleks.

Ketika tresholding diterapkan, setiap piksel dalam citra dievaluasi terhadap nilai ambang tersebut. Jika nilai piksel lebih tinggi dari ambang, maka piksel tersebut akan diubah menjadi putih (atau warna yang ditentukan), sedangkan jika lebih rendah, akan diubah menjadi hitam. Proses ini menghasilkan citra binary yang jelas, di mana objek yang relevan menjadi lebih mudah dikenali. Misalnya, saat nilai ambang ditingkatkan, objek berwarna biru dalam citra mulai muncul, diikuti oleh objek merah, dan terakhir objek hijau. Hal ini menunjukkan dengan jelas bagaimana tresholding dapat digunakan untuk meningkatkan kontras dan mempermudah analisis citra, karena informasi yang lebih relevan dapat dipisahkan dari latar belakang yang tidak diinginkan.

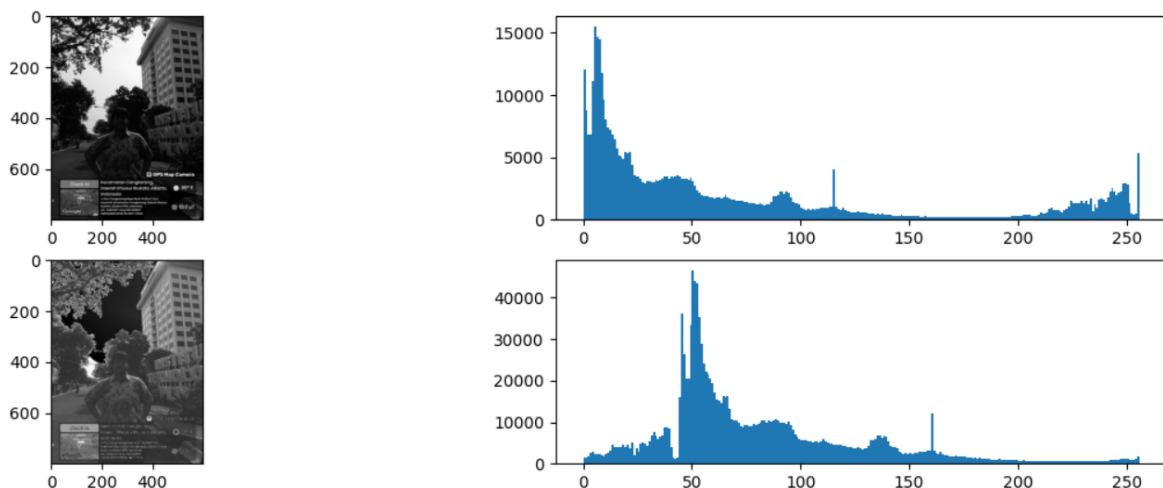


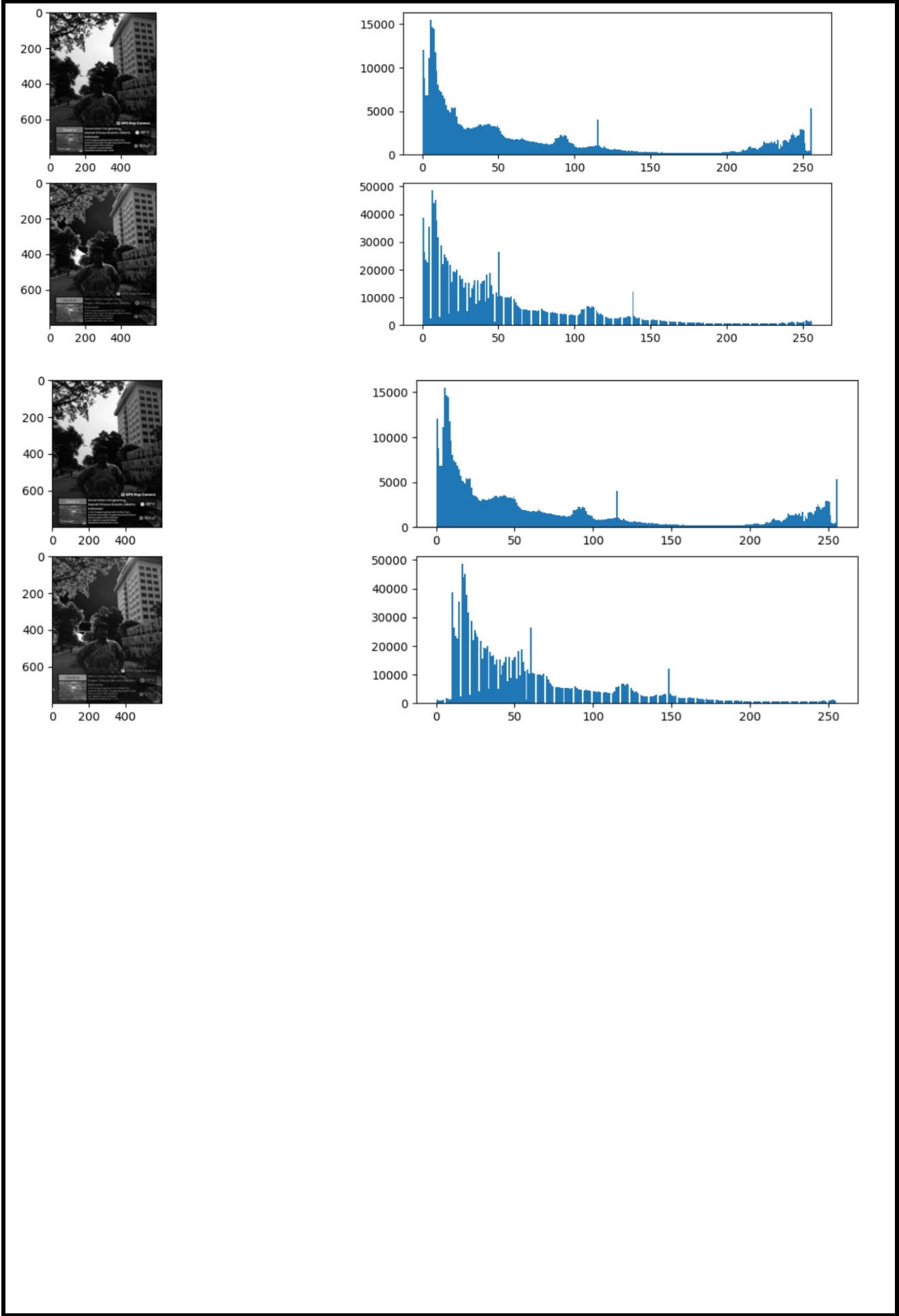


III.III. Peningkatan Citra Backlight Menjadi Citra Berkualitas Baik

Di bagian ini, saya menjelaskan bagaimana citra yang awalnya backlight diubah menjadi gambar yang baik melalui proses peningkatan kecerahan dan kontras. Citra backlight sering kali tampak gelap dan kehilangan detail, sehingga pengolahan yang tepat sangat diperlukan untuk menyajikan informasi yang jelas. Dengan menggunakan teknik peningkatan citra, saya menerapkan transformasi linear yang mengubah nilai piksel citra untuk meningkatkan kecerahan dan kontras, sehingga menghasilkan citra yang lebih cerah dan informatif.

Proses ini melibatkan penggunaan parameter kontras dan kecerahan untuk menyesuaikan citra. Dengan meningkatkan kecerahan, kami dapat membuat detail yang tersembunyi dalam bayangan menjadi lebih terlihat. Sementara itu, peningkatan kontras membantu menyoroti perbedaan antara piksel terang dan gelap, sehingga memperjelas batas objek dalam citra. Hasil akhir menunjukkan citra yang jauh lebih baik, di mana detail yang sebelumnya hilang kini terlihat jelas. Ini menegaskan bahwa teknik peningkatan kecerahan dan kontras tidak hanya meningkatkan kualitas visual citra tetapi juga memungkinkan analisis yang lebih baik terhadap informasi yang terkandung dalam citra, memberikan wawasan yang lebih mendalam bagi para peneliti dan praktisi di bidang ini.





BAB IV

PENUTUP

Berdasarkan hasil praktikum dan pembahasan yang telah dilakukan, dapat disimpulkan bahwa pengolahan citra digital merupakan proses yang sangat penting dalam dunia komputasi visual. Pemisahan channel RGB memungkinkan analisis terfokus pada masing-masing komponen warna, sementara penghilangan channel memberikan wawasan mengenai kontribusi warna tertentu terhadap keseluruhan citra. Teknik peningkatan kecerahan dan kontras terbukti efektif dalam memperjelas tampilan citra, terutama ketika kondisi awal citra memiliki pencahayaan yang buruk. Visualisasi hasil melalui Matplotlib, baik dalam bentuk citra maupun histogram, memberikan representasi yang informatif untuk analisis visual maupun statistik. Semua langkah ini memperkuat pemahaman akan pentingnya pra-pemrosesan citra dalam pipeline analisis visual.

Daftar Pustaka

- [1] B. Batchelor, "Digital Image Processing," *International Journal of Image Processing and Vision Science.*, 2022.
- [2] A. Kumar dan M. A. Professor, "Object tracking through Artificial Intelligence and Computer Vision," *NeuroQuantology*, 2023.
- [3] Y. Huang, "Overview of Research Progress of Digital Image Processing Technology," *Journal of Physics: Conference Series*, 2022.
- [4] T. Li dan H. Zhu, "Research on Color Algorithm of Gray Image Based on a Color Channel," *2020 Chinese Control And Decision Conference (CCDC)*, pp. 3747-3752, 2020.
- [5] Z. N. Khudhair, A. N. Khdiar, N. K. E. Abbadi, F. Mohamed, T. Saba, F. S. Alamri dan A. Rehman, "Color to Grayscale Image Conversion Based on Singular Value Decomposition," *IEEE Access*, vol. XI, pp. 54629-54638, 2023.
- [6] Y. Kong, C. Zhou, C. Zhang, L. Sun dan C. Zhou, "Multi-Color Channels Based Group Sparse Model for Image Restoration," *Algorithms*, vol. XV, p. 176, 2022.
- [7] G. Wang, T. Jiang, D. Zhang dan V. I. Vasil'ev, "Color image watermarking scheme based on singular value decomposition of split quaternion matrices," *J. Frankl. Inst*, p. 107508, 2025.
- [8] M. Hassan, "A uniform illumination image enhancement via linear transformation in CIELAB color space," *Multimedia Tools and Applications*, 2022.
- [9] N. Parimala, K. K. S. Teja, J. A. Kumar, G. Keerthana, K. Meghana dan R. Pitchai, "Real-time Brightness, Contrast and The Volume Control with Hand Gesture Using Open CV Python," *2024 10th International Conference on Communication and Signal Processing (ICCSP)*, 2024\.
- [10] F. Susanto dan H. S. Utami, "Analysis of Histogram and Grayscale on Chest X-Ray in Lung Cancer Using Image-J," *Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics*, 2023.
- [11] R. Tomaszewski dan J. Dajka, "Statistical image analysis and escort histograms in characterization of articular cartilage repair in a skeleton animal model," *PLoS ONE*, 2021.