

# Pembuatan Generator Seni Piksel 2-bit dengan Teknik Pengolahan Citra

Rozan Fadhil Al Hafidz - 13520039  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung  
E-mail (gmail): 13520039@std.stei.itb.ac.id

**Abstrak**—*Seni piksel (pixel art)* sudah menjadi salah satu gaya yang umum digunakan pada karya seni digital, terutama pada permainan video, animasi, dan seni digital. Makalah ini membahas pembuatan generator seni piksel 2-bit menggunakan teknik pengolahan citra. Pendekatan ini bertujuan untuk menciptakan seni piksel dengan kedalaman warna terbatas untuk merepresentasikan warna pada setiap piksel. Generator tersebut akan menerima masukan sebuah citra dan sebuah palet warna yang berisi 4 warna. Terdapat beberapa langkah yang dilakukan oleh generator, yaitu mengelompokkan warna pada citra menggunakan K-Means, menghitung jarak antar warna klaster dengan warna palet, melakukan *mode pooling* pada gambar sehingga membentuk balok-balok piksel, dan memetakan modus warna pada balok piksel dengan warna pada palet. Makalah ini diharapkan dapat memberikan wawasan baru dalam pengembangan seni piksel dengan memanfaatkan pengolahan citra untuk menciptakan gambar yang eksotis dengan warna yang terbatas.

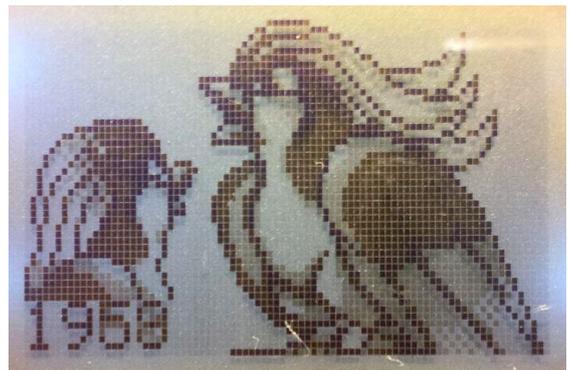
**Kata kunci**—*seni piksel; k-means; warna 2-bit; palet warna;*

## I. PENDAHULUAN

Terdapat beberapa jenis karya seni visual, seperti gambaran, lukisan, fotografi, dan sebagainya. Seiring perkembangan zaman, karya seni visual dapat dibuat dan ditampilkan menggunakan komputer. Seni piksel (*pixel art*) merupakan salah satu gaya karya seni digital yang dibangun hanya dengan menggunakan piksel (*picture element*) sebagai blok pembangun. Pada awal komputer diciptakan, seni piksel merupakan cara yang dilakukan oleh developer untuk membuat gambar digital menggunakan perangkat sumber daya komputasi yang terbatas. Pada masa tersebut, kartu grafis belum mampu untuk melakukan *render* piksel dalam jumlah besar. Oleh karena itu, *programmer* harus membuat gambar digital dengan gaya seni piksel dan memastikan gambar tersebut masuk akal [1].

Saat ini, gaya seni piksel masih digunakan oleh seniman piksel dan studio permainan video, meskipun limitasi teknologi telah dilewati. Pendekatan pembatasan warna pada seni piksel dapat memberikan pengalaman visual yang berbeda dan meningkatkan estetika klasik permainan retro. Selain itu, kemungkinan kasus penggunaan lain dari pembatasan warna adalah agar gambar seni piksel dapat ditampilkan pada perangkat yang memiliki kedalaman warna yang terbatas, misalnya pada layar kecil di perangkat IoT.

Oleh karena itu, pengembangan generator seni piksel dengan memanfaatkan teknik pengolahan citra dan membatasi kedalaman warna 2-bit menjadi tantangan menarik dalam pembuatan seni piksel. Contoh dari seni piksel 2-bit pada LCD Nokia 5110 ditunjukkan pada Gambar 1.



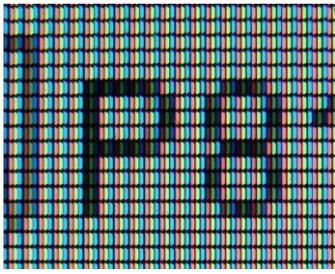
Gambar 1. Seni piksel pada LCD Nokia 5110

Makalah ini bertujuan untuk membuat generator seni piksel 2-bit. Masukan dari generator tersebut adalah sebuah citra dan sebuah palet warna 2-bit. Pengguna bisa memilih palet warna dan ukuran balok piksel yang diinginkan. Kemudian, generator akan membuat seni piksel dari citra tersebut dengan nilai dari piksel-pikselnya mengikuti palet warna yang dimasukkan. Generator tersebut dapat digunakan untuk membuat seni piksel 2-bit dengan mudah dari citra masukan yang diberikan,

## II. LANDASAN TEORI

### A. Piksel

Piksel (*picture element*) adalah unit terkecil pada tampilan digital. Setiap piksel terdiri dari upa-piksel yang memancarkan warna merah, hijau, dan biru (RGB) dengan intensitas yang berbeda. Contoh dari citra upa-piksel ditunjukkan pada Gambar 2. Kombinasi komponen warna RGB yang berbeda akan membentuk warna yang berbeda pada tampilan komputer. Jumlah piksel akan menentukan resolusi dari suatu gambar. Semakin banyak piksel, maka akan semakin tajam gambar tersebut [2].

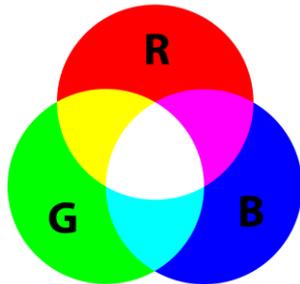


Gambar 2. Citra dari upa-piksel layar LCD yang ditangkap kamera

### B. Warna

Terdapat beberapa model warna yang dapat direpresentasikan pada komputer. Pada umumnya, model warna digunakan pada citra digital adalah model warna RGB, yaitu *red*, *green*, dan *blue*. Model warna tersebut mengikuti warna-warna yang dipancarkan oleh upa-piksel pada layar komputer. Model warna tersebut menampilkan warna-warna dengan mengombinasikan cahaya dari tiga warna (merah, biru, dan hijau). Ilustrasi dari model warna RGB ditunjukkan pada Gambar 3.

Representasi RGB juga merupakan representasi yang umum yang digunakan untuk menyimpan berkas citra digital. Citra RGB direpresentasikan dengan kedalaman 24-bit, yaitu 8 bit untuk warna merah, 8 bit untuk warna hijau, dan 8 bit untuk warna biru. Oleh karena itu, terdapat sekitar 16 juta kombinasi warna yang mungkin dibuat pada setiap piksel dan setiap piksel akan direpresentasikan dengan nilai 24 bit [3].



Gambar 3. Model warna RGB

### C. Seni Piksel

Seni piksel (*pixel art*) merupakan salah satu bentuk dari seni digital yang dibuat menggunakan perangkat lunak. Gambar pada seni piksel didesain pada level piksel. Seni piksel diciptakan pada sekitar tahun 1970-an. Pada masa tersebut, seni piksel merupakan satu-satunya jenis dari seni komputer yang dapat dibuat. Gaya visual dari seni piksel akan memiliki efek seperti seni mosaik dan manik-manik yang mengonstruksi gambar dari unit-unit warna yang lebih kecil [4].

Seni piksel umumnya menggunakan warna yang terbatas. Tahun 1983 sampai tahun 1987 merupakan era 8-bit. Setelah itu, tahun 1987 sampai tahun 1993 merupakan era 16-bit. Semakin banyak bit yang digunakan, maka akan semakin banyak warna yang ditampilkan [5]. Contoh

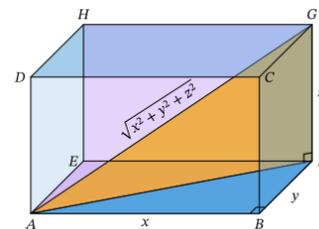
perbedaan representasi 8-bit dan 16-bit ditunjukkan pada Gambar 4.



Gambar 4. Perbedaan representasi warna 8-bit dengan 16-bit

### D. Jarak Euclidean

Pada koordinat geometri, jarak *euclidean* adalah jarak di antara dua titik. Untuk menghitung jarak *euclidean*, dapat digunakan teorema Pythagoras [6]. Ilustrasi dari perhitungan jarak *euclidean* ditunjukkan pada Gambar 5.



Gambar 5. Ilustrasi perhitungan jarak *euclidean* pada bidang 3 dimensi

Setiap piksel pada citra dengan model warna RGB direpresentasikan dengan vektor 3 dimensi. Oleh karena itu, kita bisa menghitung tingkat kemiripan dari dua buah warna dengan cara menghitung jarak *euclidean* dari kedua warna tersebut.

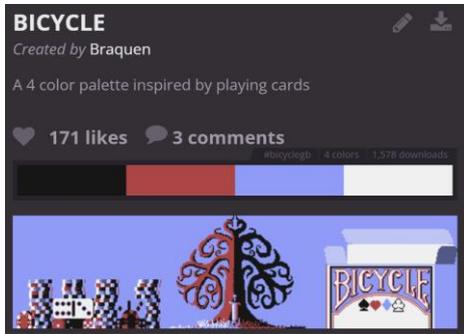
### E. Clustering

*Clustering* adalah salah satu teknik *unsupervised machine learning* untuk mengidentifikasi kelompok data yang memiliki kesamaan. Data tanpa label dapat dikelompokkan berdasarkan kesamaan fitur. Salah satu algoritma *clustering* yang populer adalah K-Means. Algoritma K-Means memiliki tujuan untuk mencari maksima lokal pada tiap iterasinya [7]. Berikut ini adalah langkah-langkah dari algoritma K-Means.

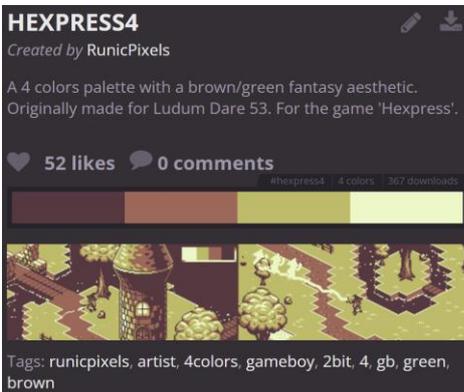
1. Tentukan jumlah kluster K. Tetapkan K buah titik sebagai pusat kluster secara acak.
2. Kelompokkan setiap data ke kluster terdekat.
3. Hitung rata-rata pada setiap kluster untuk dijadikan pusat kluster yang baru.
4. Hitung ulang setiap sampel ke pusat kluster baru terdekat
5. Ulangi langkah No. 2 sampai melewati batas iterasi yang ditentukan atau sampai tidak ada perubahan kluster

### F. Palet Warna

Palet warna (*color palette*) pada dunia digital adalah rentang warna yang dapat ditampilkan pada sebuah layar perangkat atau antarmuka lainnya. Pada kasus lain, palet warna adalah sekumpulan warna yang digunakan pada program untuk membuat karya seni visual [8]. Situs Lospec (<https://lospec.com/palette-list/tag/2bit>) menyediakan puluhan palet warna 2-bit yang dapat dijadikan referensi. Contoh dari palet warna dari situs tersebut ditunjukkan pada Gambar 6 dan Gambar 7.



Gambar 6. Palet warna Bicycle



Gambar 7. Palet warna Hexpress4

### III. DESKRIPSI SOLUSI

Pada makalah ini, dibuat generator yang akan membuat seni piksel dengan 2-bit warna dari citra yang dimasukkan. Generator tersebut juga akan memetakan warna citra terhadap palet warna yang dipilih pengguna. Selain itu, pengguna juga bisa menentukan ukuran piksel yang diinginkan untuk seni piksel yang dibuat. Bahasa yang digunakan adalah Python. Pustaka yang digunakan adalah numpy untuk membantu kalkulasi, matplotlib untuk membaca dan menyimpan gambar, sklearn untuk *K-Means clustering*, dan argparse untuk menerima argumen. Diagram alur proses yang dilakukan oleh generator ditunjukkan pada Gambar 8.



Gambar 8. Diagram alur Generator Seni Piksel 2-bit

Subbab selanjutnya akan menjelaskan langkah-langkah yang dilakukan oleh generator.

#### A. Menjalankan Program

Program menerima beberapa argumen, yaitu lokasi citra input, ukuran piksel, dan palet warna. Berikut ini adalah keluaran dari menu *help* program

```

> python main.py -h
usage: main.py [-h] [--save SAVE] image_path pixel_size
              color_palette.

Generate 2-bit pixel art from an image. Supported color
palettes are: 2bit_demichrome, bicycle, neon_night_sky,
red-blood_pain, 2_bit_matrix, b4sement, pumpkin_gb,
honey_milk, mangavania, daybreak, sunset_lattern, and
hexpress4.

positional arguments:
  image_path      Path to the image
  pixel_size      Size of each pixel
  color_palette   Color palette to use

options:
  -h, --help      show this help message and exit
  --save SAVE     Path to save the image
  
```

Berikut ini adalah isi dari beberapa contoh palet warna yang dapat digunakan.

```

"honeymilk": np.array([
    [33, 18, 16],
    [94, 45, 32],
    [199, 107, 42],
    [240, 194, 96]
]),
"b4sement": np.array([
    [34, 35, 35],
    [255, 74, 220],
    [61, 255, 152],
    [240, 246, 240]
]),
"hexpress4": np.array([
    [85, 56, 64],
    [155, 104, 89],
    [190, 188, 106],
    [237, 248, 200]
]),
"mangavania": np.array([
    [176, 62, 128],
    [255, 138, 205],
    [74, 237, 255],
    [255, 255, 255]
]),
"daybreak": np.array([
    [8, 5, 12],
    [130, 45, 48],
    [235, 117, 77],
    [248, 181, 129]
]),
"morning_mist": np.array([
    [10, 10, 36],
    [107, 99, 99],
    [167, 186, 145],
    [192, 239, 251]
]),
  
```

## B. K-Means Clustering

Dengan algoritma *K-means clustering*, program mengelompokkan warna pada citra berdasarkan kemiripan warna. Berikut ini adalah potongan kode program untuk *K-means clustering*

```
# K-Means Clustering
kmeans_model = KMeans(n_clusters = 4)
cluster_labels = kmeans_model.fit_predict(image)
rgb_cols =
kmeans_model.cluster_centers_.round(0).astype(int)
sorted_rgb_cols = sorted(rgb_cols, key=lambda c:
euclidian_distance(c, np.array([0,0,0])), reverse=True)
```

Contoh dari hasil *clustering* ditunjukkan pada Gambar 9.



Gambar 9. Hasil *clustering* dengan algoritma K-Means pada citra Lena

## C. Mode Pooling

*Mode pooling* dilakukan untuk membuat balok-balok piksel pada citra. *Mode pooling* dilakukan dengan cara mencari modus label hasil *clustering* pada tiap balok gambar. Pengguna dapat mengatur ukuran balok piksel. Berikut ini adalah potongan kode program untuk *mode pooling*.

```
# Pixelate the image using mode pooling
pixelated_cluster_labels =
cluster_labels.copy().reshape(x, y)
for i in range(x//pixel_size+1):
for j in range (y//pixel_size+1):
if i*pixel_size >= x or j*pixel_size >= y:
continue
label = np.bincount(
pixelated_cluster_labels[
i*pixel_size:(i+1)*pixel_size,
j*pixel_size:(j+1)*pixel_size
].flatten()).argmax()
pixelated_cluster_labels[
i*pixel_size:(i+1)*pixel_size,
j*pixel_size:(j+1)*pixel_size
] = label
```

Contoh hasil dari *mode pooling* dengan beberapa ukuran piksel ditunjukkan pada Gambar 10.



Gambar 10. Hasil *model pooling* pada citra Lena

## D. Pemetaan Warna

Setelah itu, dilakukan pemetaan warna dari warna hasil *clustering* terhadap warna pada palet warna. Algoritma yang digunakan adalah sebagai berikut

1. Urutkan warna berdasarkan jumlah nilainya
2. Buat pemetaan untuk warna sebelum diurutkan terhadap warna setelah diurutkan
3. Buat pemetaan untuk warna hasil *clustering* terhadap warna pada palet warna dengan cara menghitung jarak *euclidean*-nya dari warna tercerah.
4. Aplikasikan pemetaan dari citra hasil *mode pooling* dengan pemetaan yang sudah dibuat

Berikut ini adalah potongan kode program untuk pemetaan warna.

```
# Get the index of the sorted colors
idx_not_sorted_to_sorted = []
for i in range(len(rgb_cols)):
idx_not_sorted_to_sorted.append(
np.where(np.all(
sorted_rgb_cols == rgb_cols[i], axis=1
))[0][0])

# Map the sorted colors to the palette
idx_sorted_to_palette = []
temp_color_palete = color_palette.copy()
for i in range(len(color_palette)):
idx_sorted_to_palette.append(
sorted(
temp_color_palete,
key=lambda c: euclidian_distance(
c, sorted_rgb_cols[i]
)
)[0])
temp_color_palete = [
c for c in temp_color_palete if
not np.array_equal(c,
idx_sorted_to_palette[i])
]
```

Contoh hasil dari pemetaan dengan beberapa palet warna ditunjukkan pada Gambar 11.



Gambar 11. Hasil pemetaan warna pada citra Lena dengan ukuran piksel 4

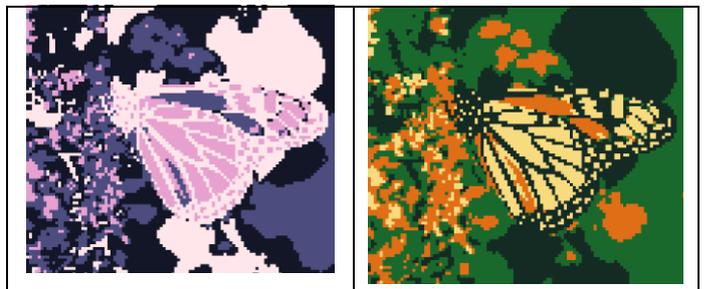
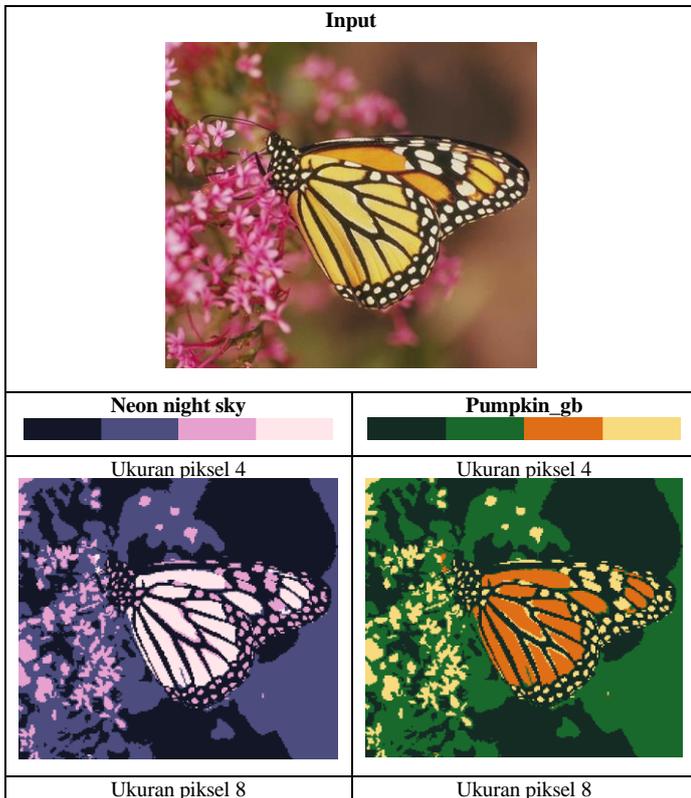
#### IV. HASIL DAN PENGUJIAN

Pada bab ini akan dilakukan pengujian program menggunakan beberapa citra lain.

##### A. Kupu-Kupu

Berikut ini adalah pengujian dengan menggunakan citra kupu-kupu

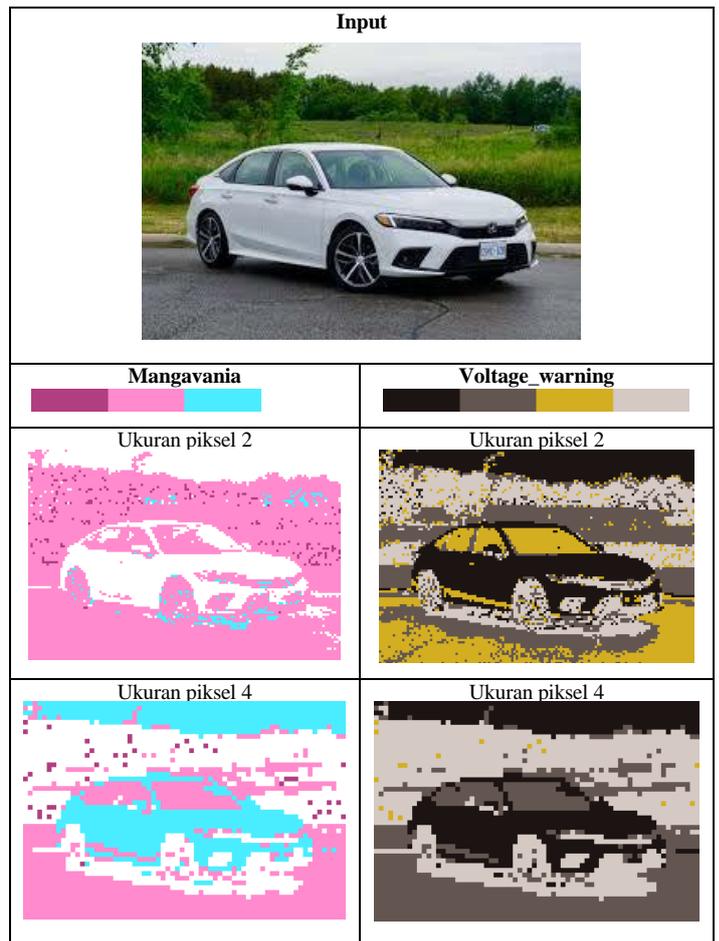
TABEL I. PENGUJIAN DENGAN CITRA KUPU-KUPU



##### B. Mobil

Berikut ini adalah pengujian dengan menggunakan citra mobil

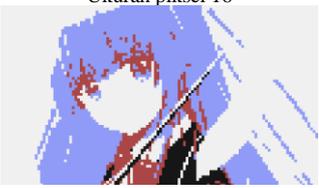
TABEL II. PENGUJIAN DENGAN CITRA MOBIL



##### C. Anime

Berikut ini adalah pengujian dengan menggunakan gambar anime.

TABEL III. PENGUJIAN DENGAN GAMBAR ANIME

Input	
	
<b>Bicycle</b> 	<b>B4sement</b> 
Ukuran piksel 8 	Ukuran piksel 8 
Ukuran piksel 16 	Ukuran piksel 16 

## V. KESIMPULAN DAN SARAN

Telah dibuat sebuah generator seni piksel 2-bit dengan menggunakan teknik pengolahan citra. Teknik-teknik yang digunakan adalah *K-means clustering*, *pooling*, dan perhitungan jarak *euclidean* pada model warna RGB. Generator tersebut bekerja dengan baik pada citra yang tidak banyak mengandung warna utama. Namun, jika citra input mengandung banyak utama, hasil dari generator tersebut akan kurang menggambarkan citra aslinya. Hal tersebut karena 2-bit warna tidak dapat merepresentasikan citra yang memiliki lebih dari 4 warna utama.

PRANALA VIDEO DI YOUTUBE

<https://youtu.be/jG4Z3ZWynVw>

PRANALA GITHUB

<https://github.com/rozanfa/2bit-pixel-art-generator>

UCAPAN TERIMA KASIH

Penulis mengucapkan puji syukur kepada Tuhan YME karena atas rahmat dan karunia-Nya, makalah yang berjudul "Pembuatan Generator Seni Piksel 2-bit dengan Teknik Pengolahan Citra" dapat diselesaikan tepat pada waktunya

sebagai salah satu tugas dari mata kuliah IF4073 Interpretasi dan Pengolahan Citra. Penulis juga mengucapkan terima kasih kepada Bapak Dr. Ir. Rinaldi, M.T. selaku dosen mata kuliah tersebut yang telah membimbing penulis dalam memahami materi yang digunakan dalam pembuatan makalah ini. Terakhir, penulis juga mengucapkan terima kasih kepada kedua orang tua dan keluarga yang mendukung penulis dalam mengerjakan makalah ini.

## REFERENCES

- [1] M. Rouse, "What is Pixel Art? - Definition from Techopedia," *Techopedia*, 19 Juni 2023. <https://www.techopedia.com/definition/8884/pixel-art> (diakses pada 18 Desember 2023).
- [2] A. S. Gillis, "pixel," *WhatIs*, 30 Agustus 2022. <https://www.techtarget.com/whatis/definition/pixel> (diakses pada 18 Desember 2023).
- [3] "Digital Imaging Tutorial - Basic Terminology." <http://preservationtutorial.library.cornell.edu/tutorial/intro/intro-04.html> (diakses pada 18 Desember 2023).
- [4] [1]R. Wisznia, "Pixel Art or the art of fun and original creation," *Quark*, 09 Desember 2023. <https://get-quark.com/en/blogs/createur-d-interactions/pixel-art> (diakses pada 18 Desember 2023).
- [5] S. Barnes, "How the Humble Pixel Became a Building Block To Groundbreaking Art," *My Modern Met*, 23 Maret 2022. <https://mymodernmet.com/what-is-pixel-art/> (diakses pada 18 Desember 2023).
- [6] Nagwa, "Lesson Explainer: Points, Midpoints, and Distances in Space | Nagwa," *Nagwa*. <https://www.nagwa.com/en/explainers/945187689535/> (diakses pada 18 Desember 2023).
- [7] S. Kaushik, "Clustering | Introduction, Different Methods, and Applications (Updated 2023)," *Analytics Vidhya*, 11 Juli 2023. <https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering> (diakses pada 18 Desember 2023).
- [8] M. Rouse, "What Does Color Palette Mean? - Definition from Techopedia," *Techopedia*, 20 September 2023. <https://www.techopedia.com/definition/6547/color-palette> (diakses pada 19 Desember 2023).

## PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 19 November 2023



Rozan Fadhil Al Hafidz  
13520039