



TUGAS AKHIR - KM184801

PENERAPAN METODE *COLOR INVARIANT* UNTUK PENGHAPUSAN BAYANGAN PADA CITRA DIGITAL

SASKIA RIZKIANA PUTRI
06111640000015

Dosen Pembimbing
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

Departemen Matematika
Fakultas Sains dan Analitika Data
Institut Teknologi Sepuluh Nopember
Surabaya 2020



TUGAS AKHIR– KM184801

**PENERAPAN METODE COLOR INVARIANT UNTUK
PENGHAPUSAN BAYANGAN PADA CITRA DIGITAL**

**SASKIA RIZKIANA PUTRI
NRP 06111640000015**

**Dosen Pembimbing :
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT**

**DEPARTEMEN MATEMATIKA
Fakultas Sains dan Analitika Data
Institut Teknologi Sepuluh Nopember
Surabaya 2020**



FINAL PROJECT– KM184801

***APPLICATION OF COLOR INVARIANT METHOD FOR
SHADOW REMOVAL BASED ON DIGITAL IMAGE***

**SASKIA RIZKIANA PUTRI
NRP 06111540000015**

**Supervisor :
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT**

**DEPARTMENT OF MATHEMATICS
Faculty of Mathematics, Computing, and Data Science
Sepuluh Nopember Institute of Technology
Surabaya 2020**

**LEMBAR PENGESAHAN
PENERAPAN METODE COLOR INVARIANT UNTUK
PENGHAPUSAN BAYANGAN PADA CITRA DIGITAL**

***APPLICATION OF COLOR INVARIANT METHOD FOR
SHADOW REMOVAL BASED ON DIGITAL IMAGE***


TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat
Untuk memperoleh gelar Sarjana Matematika
Pada bidang studi Ilmu Komputer
Program Studi S-1 Departemen Matematika
Fakultas Sains dan Analitika Data
Institut Teknologi Sepuluh Nopember Surabaya
Oleh :

Saskia Rizkiana Putri
NRP. 06111640000015

Menyetujui,

Dosen Pembimbing I,



Dr. Dwi Ratna Sulistyanningrum, S.Si, MT

NIP. 19690405 199403 2 003

Mengetahui,

Kepala Departemen Matematika

FSAD ITS



Subchan, Ph.D

NIP. 19710513 199702 1 001

Surabaya, 19 Agustus 2020

PENERAPAN METODE COLOR INVARIANT UNTUK PENGHAPUSAN BAYANGAN PADA CITRA DIGITAL

Nama : Saskia Rizkiana Putri
NRP : 06111640000015
Departemen : Matematika FSAD - ITS
Pembimbing : Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

ABSTRAK

Bayangan terjadi ketika cahaya dari sebuah sumber cahaya terhalangi oleh sebuah objek. Pada pengolahan citra digital bayangan dapat menyebabkan banyak kendala dan kesalahan informasi. Terlebih saat ada sebuah objek yang terhalangi oleh bayangan. Hal tersebut dapat menyebabkan kesalahan dalam proses deteksi atau pengenalan objek. Salah satu solusi dalam mengatasi permasalahan tersebut adalah dengan melakukan proses reduksi bayangan. Metode *Color Invariant* digunakan untuk menyelesaikan permasalahan tersebut. Langkah pertama dari metode ini adalah menentukan kromatisitas log dari citra *input*. Selanjutnya adalah mendapatkan citra *illumination invariant* 1D dengan menggunakan minimasi entropi. Setelahnya akan dihasilkan citra kromatisitas dimana bayangan sudah tereduksi. Pada tugas akhir ini, citra kromatisitas diubah menjadi citra *grayscale* untuk memperoleh nilai RMSE. Namun, hasil citra output masih kurang bagus dari sisi visual karena tingkat kecerahan yang berbeda dengan citra *input* tanpa bayangan. Nilai RMSE tertinggi yang didapat adalah 0,361 dan terendah 0,013.

Kata Kunci: Bayangan, *Color Invariant*, entropi, *Root Mean Square Error* (RMSE).

APPLICATION OF COLOR INVARIANT METHOD FOR SHADOW REMOVAL BASED ON DIGITAL IMAGE

Name : Saskia Rizkiana Putri
NRP : 06111640000015
Department : Mathematics FSAD - ITS
Supervisor : Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

ABSTRACT

Shadow occurs when light from a light source is blocked by an object. In digital image processing, shadows can cause many obstacles and misinformation. Especially when there is an object that is blocked by the shadow. This can cause errors in the process of object detection or recognition. One solution to overcome these problems is to do the process of shadow reduction. The Color Invariant method is used to solve this problem. The first step of this method is to determine the log chromaticity of the input image. Next is to get an 1D invariant illumination image using entropy minimization. After that a chromaticity image will be produced where the image has been reduced. In this final project, the chromaticity image is converted to a grayscale image to obtain an RMSE value. However, the output image results are still not good from the visual side because the brightness is different from the input image without shadow. The highest RMSE value obtained was 0.361 and the lowest was 0.013.

Keywords: Shadow, Color Invariant, Entropy , Root Mean Square Error (RMSE).

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Alhamdulillahahirabbil'aalamiin, segala puji dan syukur penulis panjatkan ke hadirat Allah SWT. yang telah memberikan limpahan rahmat, taufik dan hidayah – Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul Penerapan Metode Color Invariant untuk Penghapusan Bayangan pada Citra Digital sebagai salah satu syarat kelulusan Program Sarjana Departemen Matematika FSAD Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Tugas Akhir ini dapat terselesaikan dengan baik dan tepat waktu berkat bantuan serta dukungan dari banyak pihak. Oleh karena itu, penulis menyampaikan ucapan terimakasih kepada :

1. Keluarga penulis, pertama kepada kedua orang tua Bapak Joko Sumaryadi dan Ibu Triana Wijaya Kusumawati yang setia mendampingi dan memberi semangat sejak penulis lahir hingga saat ini. Tanpa doa restu beliau, penulis tidak akan mampu berada di titik ini. Kedua kepada adik saya Annisa, yang selalu menjadi tempat untuk bercerita tentang keluh kesah dari penulis.
2. Bapak Subchan, Ph.D sebagai Kepala Departemen Matematika FSAD ITS.
3. Ibu Dr. Dwi Ratna Sulistyaningrum, S.Si, MT selaku dosen pembimbing yang telah memberikan arahan, nasehat serta saran dengan sangat sabar sehingga penulis bisa menyelesaikan Tugas Akhir ini dengan baik dan lancar.
4. Ibu Dra. Wahyu Fistia Doctorina, M.Si selaku dosen wali penulis.
5. Bapak dan Ibu Dosen serta Staff Departemen Matematika ITS.

6. Rida , Renita, Sandy dan Dita selaku sahabat yang selalu menjadi pendengar penulis saat berkeluh kesah.
7. Kharisma, Merti, Shinta, Nuchan dan Alvaro, selaku tim satu pengolahan cita , tempat berbagi suka dan duka tentang citra digital.
8. LEMNISCATE kawan seangkatan yang selalu berbagi kisah senang maupun duka. Kawan yang selalu saling membantu dan solid.

Penulis menyadari bahwa dalam Tugas Akhir ini masih terdapat kekurangan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan. Akhirnya penulis berharap semoga Tugas Akhir ini dapat bermanfaat bagi banyak pihak.
Wassalamu'alaikum Wr. Wb.

DAFTAR ISI

HALAMAN JUDUL.....	iii
TITLE PAGE	v
LEMBAR PENGESAHAN....	Error! Bookmark not defined.
ABSTRAK.....	ix
<i>ABSTRACT</i>	xi
KATA PENGANTAR	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL.....	xix
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah	3
1.4. Tujuan.....	3
1.5. Manfaat	4
1.6. Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA	7
2.1. Penelitian Terdahulu	7
2.2. Landasan Teori	9
2.2.1. Pengertian Citra Digital.....	9
2.2.5. Bayangan.....	14
2.2.6. <i>Color Invariant</i>	16
2.2.7. Entropi.....	25
2.2.8. <i>Root Mean Square Error</i> (RMSE).....	26

BAB III METODE PENELITIAN	29
3.1. Data Penelitian	29
3.2. Tahapan Penelitian	29
 BAB IV PERANCANGAN DAN IMPLEMENTASI PROGRAM.....	35
4.1. Perancangan Program	35
4.2. Implementasi Program	47
4.3. Tampilan <i>Interface</i>	56
 BAB V UJI COBA DAN ANALISA HASIL	59
5.1. Data Uji Coba	59
5.2. Hasil Uji Coba	70
5.2.1 <i>Background</i> citra bayangan cenderung abu-abu dan hitam	70
5.2.2 <i>Background</i> citra bayangan berwarna	72
5.2.3 Citra bayangan dengan objek	75
5.2.4 Citra bayangan dengan objek pembentuknya	78
5.2.5 Citra bayangan dengan objek warna mirip bayangan ..	80
5.2.6 Citra bayangan pada kerusakan jalan	83
 BAB VISIMPULAN DAN SARAN	87
6.1. Kesimpulan	87
6.2. Saran	88
 DAFTAR PUSTAKA	89
LAMPIRAN.....	93
TENTANG PENULIS	148

DAFTAR GAMBAR

Gambar 2. 1	Piksel pada sebuah citra [6]	10
Gambar 2. 2	Ilustrasi digitalisasi citra [6].....	11
Gambar 2. 3	Citra biner [6]	11
Gambar 2. 4	Citra <i>grayscale</i> [6]	12
Gambar 2. 5	piksel format 8 bit.....	13
Gambar 2. 6	citra warna (a) 8 bit (b) 16 bit (c) 24 bit [6]	13
Gambar 2. 7	ilustrasi <i>self shadow</i> dan <i>cast shadow</i> [7].....	14
Gambar 2. 8	ilustrasi <i>umbra</i> dan <i>penumbra</i> [7]	15
Gambar 3. 1	Langkah-langkah penelitian	30
Gambar 3. 2	Diagram Alir Penghapusan Bayangan	32
Gambar 4.1	Bayangan Manusia 1600x 1200 piksel.....	35
Gambar 4.2	Citra input 200 x 200 piksel.....	36
Gambar 4. 3	Citra <i>input</i> 640x 480 piksel.....	48
Gambar 4. 4	Citra <i>resize</i> 200 x 200 piksel.....	49
Gambar 4. 5	(a) Citra <i>input channel Red</i> , (b) Citra <i>input channel Green</i> , (c) Citra <i>input channel Blue</i>	49
Gambar 4. 6	Citra <i>input illumination invariant</i> 1D.....	52
Gambar 4. 7	(a) Citra <i>input</i> , (b) Citra <i>illumination invariant</i> 1D , (c) Citra Kromatisitas.....	53
Gambar 4. 8	Tampilan interface	56
Gambar 5. 1	Perbandingan citra <i>input grayscale</i>	72
Gambar 5. 2	Perbandingan citra <i>input grayscale</i>	74
Gambar 5. 3	Perbandingan citra <i>input</i>	74
Gambar 5. 4	Perbandingan citra <i>input grayscale</i>	76
Gambar 5. 5	Perbandingan citra <i>input grayscale</i>	77
Gambar 5. 6	Perbandingan citra asli dengan bayangan <i>grayscale</i>	77
Gambar 5. 7	Perbandingan citra <i>input grayscale</i>	79
Gambar 5. 8	Perbandingan citra <i>input grayscale</i>	81

Gambar 5. 9	Perbandingan citra <i>input grayscale</i>	81
Gambar 5. 10	Perbandingan citra <i>input grayscale</i>	82
Gambar 5. 11	Perbandingan citra <i>input grayscale</i>	84
Gambar 5. 12	Perbandingan citra <i>input grayscale</i>	85

DAFTAR TABEL

Tabel 5. 1.	Data Uji Coba.....	60
Tabel 5. 2	Hasil ketiga RMSE pada citra bayangan dengan <i>background</i> cenderung abu-abu dan hitam	71
Tabel 5. 3	Hasil ketiga RMSE pada citra bayangan dengan <i>background</i> berwarna	72
Tabel 5. 4	Hasil ketiga RMSE pada citra bayangan dengan objek.....	75
Tabel 5. 5	Hasil ketiga RMSE pada citra bayangan dengan objek pembentuknya.....	78
Tabel 5. 6	Hasil ketiga RMSE pada citra bayangan dengan objek warna mirip bayangan.....	80
Tabel 5. 7	Hasil ketiga RMSE pada citra bayangan pada kerusakan jalan.....	83
Tabel 5. 8	Perbandingan Citra <i>input</i> , Citra asli tanpa bayangan, dan Citra <i>output</i>	86

BAB I

PENDAHULUAN

Bab ini membahas latar belakang secara umum yang mendasari penulisan Tugas Akhir mengenai penghapusan bayangan dengan menggunakan metode *Color Invariant* pada citra digital. Kemudian dijabarkan juga rumusan masalah, batasan masalah, tujuan, dan manfaat yang diambil berdasarkan latar belakang penyusunan Tugas Akhir ini.

1.1. Latar Belakang

Pada era sekarang ini, mengabadikan sebuah peristiwa adalah hal yang seakan wajib untuk dilakukan. Kebanyakan orang mengabadikan peristiwa berharga mereka dengan menggunakan kamera. Kamera adalah suatu alat yang dapat menangkap serta merekam momen dalam bentuk gambar maupun video. Saat ini Kamera bukanlah hal yang susah untuk didapatkan dan bahkan sekarang semua *smartphone* pun sudah dilengkapi dengan kamera. Hal tersebut meningkatkan kemungkinan banyak nya peristiwa yang terekam dalam bentuk foto atau video.

Bayangan merupakan salah satu jenis penghalang dalam pengambilan gambar. Bayangan dapat membuat hasil foto menjadi kurang optimal karena objek foto tertutupi bagian gelap dari bayangan. Akibatnya, informasi yang seharusnya bisa didapatkan dari foto tersebut menjadi kurang valid. Padahal dari sebuah foto, bisa didapat informasi yang sangat berguna. Foto dapat diolah untuk selanjutnya dilakukan deteksi, identifikasi dan klasifikasi objek. Salah satu permasalahan nyata yang terjadi adalah bayangan yang menghalangi kerusakan jalan yang berupa retakan dan lubang. Apabila foto kerusakan jalan tersebut langsung diolah untuk proses identifikasi maupun klasifikasi jenis kerusakan jalan

maka akibatnya hasil akurasi kurang optimal dan bisa terjadi kesalahan dalam pengklasifikasian. Oleh karena itu dibutuhkan suatu media untuk mengurangi atau bahkan menghilangkan bayangan yang tertangkap oleh kamera pada saat pengambilan foto.

Sampai saat ini, sudah ada beberapa penelitian yang membahas tentang penghapusan bayangan pada foto. Pada tahun 2013, Qiang He dkk. melakukan penelitian penghapusan bayangan dengan menggunakan *K-Means Method* dan *Fisher Linear Discriminant*[1]. Selanjutnya pada tahun 2014, Zhang Yali dkk. juga melakukan penelitian penghapusan bayangan dengan menggunakan metode lain dengan menerapkan *Histogram Matching* dan *Color Model Recovery* [2]. Sedangkan pada tahun 2017, Abbas Nasrabadi dan Mahdi Vaezi melakukan penelitian lainnya dengan menggunakan *Gabor Texture Filtering* dan *Color Segmentation* [3].

Color invariant adalah metode terbaru tentang pengenalan berbasis warna menemukan deskriptor invarian warna (*color invariant*) dari distribusi warna tanpa menggunakan algoritma ketetapan warna (*color constancy*) *full-blown*. *Color invariant* tidak memperkirakan penggambaran cahaya dan pantulan permukaan secara langsung. Sebaliknya mereka merupakan fitur yang tidak berubah dengan perubahan cahaya. Ada berbagai cara untuk mendapatkan fitur invarian antara lain : menormalkan lokasi citra dengan referensi RGB, dengan menggunakan rasio warna , dan hue [4].

Berdasarkan penjelasan diatas, pada tugas akhir ini dilakukan penelitian mengenai penghapusan bayangan pada citra digital dengan menggunakan metode *Color Invariant*.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas, permasalahan yang akan dibahas pada penelitian Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana proses metode *Color Invariant* untuk menghapus bayangan pada citra digital?
2. Bagaimana kinerja dan waktu komputasi metode *Color Invariant* untuk menghapus bayangan pada citra digital?

1.3. Batasan Masalah

Dalam Penelitian Tugas Akhir ini, penulis membatasi permasalahan sebagai berikut :

1. *Input* berupa citra berwarna RGB dengan format .jpg dan .png.
2. Data yang digunakan adalah dataset citra yang didapat dari pengambilan oleh penulis sendiri dan *github*.
3. Bayangan yang akan dihapus adalah *cast shadow* yaitu area yang diproyeksikan oleh objek yang searah dengan cahaya langsung.

1.4. Tujuan

Tujuan dari penelitian Tugas Akhir ini adalah sebagai berikut :

1. Menerapkan proses *Color Invariant* dalam menghapus bayangan pada citra digital.
2. Menganalisis kinerja *Color Invariant* dalam menghapus bayangan pada citra digital.

1.5. Manfaat

Dari penelitian ini diharapkan akan memberikan manfaat sebagai berikut :

1. Membantu penelitian yang terkait dengan penghapusan bayangan pada citra digital.
2. Sebagai langkah pra pengolahan untuk proses identifikasi, klasifikasi dan pengenalan objek berbasis citra digital.

1.6. Sistematika Penulisan

Penulisan Tugas Akhir ini disusun dalam lima bab, yaitu:

1. BAB I : PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat penelitian penghapusan bayangan pada citra digital dan sistematika penulisan Tugas Akhir.

2. BAB II : TINJAUAN PUSTAKA

Bab ini menjelaskan teori dasar yang mendukung penulisan Tugas Akhir, yaitu penelitian terdahulu, citra digital, Bayangan, *Color Invariant*, dan *Root Mean Square Error* (RMSE).

3. BAB III : METODE PENELITIAN

Bab ini menjelaskan tentang tahapan-tahapan dan metode yang *Color Invariant* disertai penjelasan dalam tiap tahapan yang dilakukan dalam menyelesaikan Tugas Akhir.

4. BAB IV : PERANCANGAN DAN IMPLEMENTASI PROGRAM

Bab ini membahas tentang perancangan program, yaitu penjelasan kegiatan yang dilakukan sesuai dengan yang dijabarkan pada metode penelitian, seperti bagaimana penggunaan *Color Invariant* pada Tugas

Akhir ini, dan implementasi program tersebut ke dalam bahasa komputer.

5. **BAB V : UJI COBA DAN ANALISA HASIL**

Bab ini menjabarkan data yang digunakan untuk uji coba dan hasil serta Analisa penghapusan bayangan pada citra digital yang dilakukan pada Tugas Akhir

6. **BAB VI : SIMPULAN DAN SARAN**

Bab ini menjelaskan kesimpulan akhir yang diperoleh dari uji coba dan analisa hasil serta saran untuk pengembangan penelitian selanjutnya.

BAB II

TINJAUAN PUSTAKA

Pada bab ini diuraikan mengenai penelitian terdahulu terkait penghapusan bayangan, serta teori tentang citra digital, pengertian dari bayangan, *Color Invariant* dan *Root Mean Square Error* (RMSE).

2.1. Penelitian Terdahulu

Penelitian mengenai penghilang bayangan pada sebuah foto atau citra bukanlah penelitian yang baru saja dilakukan. Beberapa peneliti terdahulu telah melakukan penelitian seperti ini.

Pada tahun 2013, Qiang He dkk. melakukan penelitian dengan judul “*A New Shadow Removal Method for Color Image*” yang menggunakan *K-Means Method* dan *Fisher Linear Discriminant*. Pada penelitian ini inputan yang digunakan berupa citra berwarna dengan bayangan. Pertama akan dihitung peta kromatisitas log dua dimensi dari gambar warna. Kedua, *Cluster* data dalam ruang kromatisitas log dua dimensi menggunakan algoritma *K-Means*. Ketiga, hitung arah invariant (garis proyeksi) menggunakan *Fisher Linear Discriminant*. Keempat, hasilkan gambar invariant dengan memproyeksikan data dua dimensi sepanjang arah invariant. Kelima, hitung peta tepi untuk gambar abu-abu (setiap *channel* R, G, B) dari gambar asli dan gambar invariant tingkat abu-abu menggunakan *Canny Edge Detectors*. Keenam, ekstrak tepi tersebut pada peta tepi gambar asli tapi tidak pada peta tepi gambar invariant sebagai tepi bayangan. Ketujuh, Hitung output filter turunan untuk *channel* R, G, dan B. selanjutnya, tetapkan nilai derivative pada posisi *edge shadow* ke nol.

Terakhir, reintegrasikan kembali output filter turunan untuk menutupi gambar bebas bayangan [1].

Pada tahun 2014, Zhang Yali dkk. juga melakukan penelitian dengan judul “*Shadow Removal Of Single Texture Region Using Histogram Matching and Color Model Recovery*”. Penelitian ini dibagi menjadi dua tahap proses yaitu deteksi bayangan dan penghapusan bayangan. Pada tahap deteksi bayangan, bayangan pada citra berwarna akan dikenali dengan menggunakan *Histogram Matching*. Output citra setelah *Histogram Matching* adalah citra biner yang menampilkan bagian tepi dari bayangan. Setelahnya citra akan diolah dengan *HSI Color Model Recovery* untuk proses penghapusan bayangan [2].

Pada tahun 2017, Abbas Nasrabadi dan Mahdi Vaezi melakukan penelitian dengan judul “*Shadow Removal using Texture and Color Clustering*”. Penelitian ini akan dilakukan *clustering* tekstur dan warna. Setelah itu akan dihitung kecerahan rata-rata dari kedua kelompok *cluster*. Pada penelitian ini juga menerapkan konversi LAB untuk mengukur kecerahan yang lebih nyaman. Selain konversi LAB, *Gabor Texture Filtering* dan *color segmentation* juga diterapkan. *Color segmentation* yang dilakukan disini adalah dengan mencari komponen RGB serupa di seluruh citra (gambar). Setelahnya akan didapatkan citra warna tanpa bayangan [3] .

Color Invariant tidak hanya digunakan untuk penghilangan bayangan. Pada tahun 2014, Shraddha Singhal et.al melakukan penelitian yang berjudul “*Automatic Detection of Bulidings from Aerial Images Using Color Invariant Features and Canny Edge Detection* ”. Penelitian ini bertujuan untuk segmentasi bangunan dari gambar (citra) udara [5].

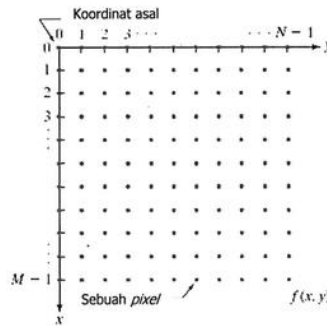
2.2. Landasan Teori

Sebelum membahas lebih dalam mengenai penghapusan bayangan pada citra digital, perlu diketahui beberapa teori dasar yang berhubungan dengan penelitian ini, seperti teori tentang citra digital, citra biner, citra *grayscale*, citra RGB, bayangan, *Color Invariant* dan RMSE. Berikut merupakan penjabaran dari teori-teori tersebut.

2.2.1. Pengertian Citra Digital

Secara umum, pengolahan citra digital menunjuk pada pemrosesan gambar dua dimensi menggunakan computer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data dua dimensi. Citra digital merupakan larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu [6].

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y sebagai koordinat spasial, dan amplitude f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y, dan nilai amplitude f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Piksel pada citra digital dapat dilihat pada Gambar 2.1.

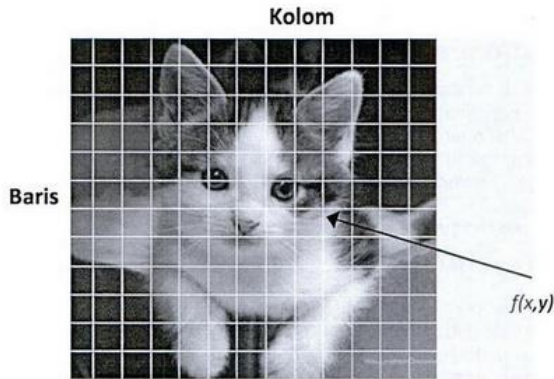


Gambar 2. 1 Piksel pada sebuah citra [6]

Citra digital dapat ditulis dalam bentuk matrik sebagai berikut

$$F = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan *picture elements*, *image elements*, *pels*, atau *pixels*. Istilah terakhir (*pixel*) paling sering digunakan pada citra digital. Ilustrasi citra digital dapat dilihat pada Gambar 2.2.



Gambar 2. 2 Ilustrasi digitalisasi citra [6]

2.2.2 Citra Biner

Citra biner merupakan citra digital yang hanya memiliki dua nilai piksel yaitu hitam dan putih [6]. Sebutan lain untuk citra Biner adalah B&W(*black and white*) atau citra monokrom. Setiap nilai piksel dari citra biner diwakilkan hanya dengan 1 bit. Citra biner juga sering muncul sebagai hasil dari proses segmentasi citra.



Gambar 2. 3 Citra biner [6]

2.2.3 Citra *Grayscale*

Citra *grayscale* adalah citra digital dengan satu nilai *channel* pada setiap piksel nya [6]. *Channel* yang dimaksudkan disini adalah RED, GREEN dan BLUE. Pada citra *grayscale* nilai dari RED = GREEN = BLUE. Nilai tersebut akan menunjukkan tingkat intensitas. Warna yang dimiliki adalah warna hitam, keabuan, dan putih. Untuk warna keabuan memiliki berbagai tingkatan dari hitam hingga mendekati putih. Citra *grayscale* berikut memiliki kedalaman warna 8 bit.



Gambar 2. 4 Citra *grayscale* [6]

2.2.4 Citra Warna

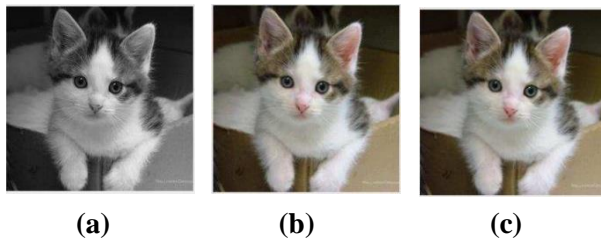
Citra warna memiliki beberapa variasi ukuran bit, antara lain 8 bit, 16 bit dan 24 bit [6]. Untuk citra warna 8 bit diwakili oleh 8 bit dengan jumlah warna maksimum adalah 256 warna. Terdapat dua jenis citra warna 8 bit. Pertama, dengan menggunakan palet warna 256 yang setiap paletnya memiliki pemetaan nilai RGB tertentu. Kedua, setiap piksel memiliki format 8 bit seperti pada gambar 2.5.

Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
R	R	R	G	G	G	B	B

Gambar 2. 5 piksel format 8 bit

Citra warna 16 bit setiap pikselnya diwakili dengan 2 *byte memory* (16 bit). Warna 16 bit memiliki 65.536 warna. Dalam formasi bitnya, nilai RED dan BLUE berada pada 5 bit di kanan dan kiri. Nilai GREEN memiliki 5 bit ditambah 1 bit tambahan.

Citra warna 24 bit diwakili dengan 24 bit dengan total 16.777.216 variasi warna. Setiap poin informasi piksel (RGB) disimpan dalam 1 *byte* data. Dengan rincian, 8 bit pertama menyimpan nilai BLUE, diikuti nilai GREEN pada 8 bit kedua dan pada 8 bit terakhir adalah nilai RED.



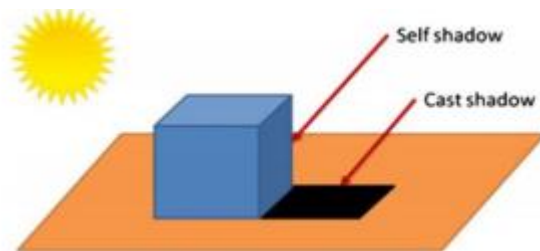
Gambar 2. 6 citra warna (a) 8 bit (b) 16 bit (c) 24 bit [6]

2.2.5. Bayangan

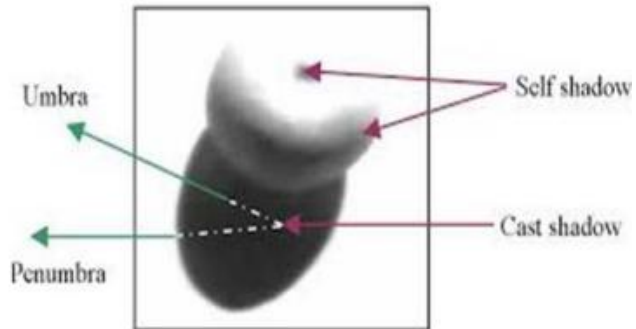
Bayangan terjadi ketika suatu objek sebagian atau seluruhnya menghalangi cahaya langsung dari sumber penerangan [7]. Bayangan dapat dibagi menjadi dua kelas yaitu *self shadow* dan *cast shadow*. *Self shadow* dan *cast shadow* memiliki tingkat kecerahan yang berbeda.

Self shadow merupakan daerah bayangan pada objek itu sendiri. *Self shadow* juga biasanya memiliki tingkat kecerahan yang lebih tinggi daripada *cast shadow* karena bagian tersebut mendapatkan lebih banyak pencahayaan sekunder dari benda-benda disekitarnya.

Cast shadow terdiri dari dua bagian yaitu umbra dan penumbra. *Umbra* adalah bagian dari *cast shadow* dimana cahaya langsung sepenuhnya diblokir atau terhalangi oleh objek. Sedangkan *penumbra* adalah bagian dari *cast shadow* dimana cahaya langsung hanya terhalangi secara parsial atau sebagian.



Gambar 2. 7 ilustrasi *self shadow* dan *cast shadow* [7]



Gambar 2. 8 ilustrasi *umbra* dan *penumbra* [7]

Karakteristik bayangan menurut Yongzhen, dkk. (2014) berisi banyak informasi dalam gambar digital [8]. Ada banyak petunjuk visual penting dari bayangan seperti kedalaman, bentuk, isi, dan pencahayaan yang dideskripsikan sebagai berikut :

- a. Nilai piksel dari bayangan harus rendah dalam semua warna RGB. Bayangan secara general lebih gelap dari daerah di sekitarnya.
- b. Bayangan tidak mengubah secara signifikan baik warna atau tekstur permukaan gambar yang ditutupi. Penutupan permukaan cenderung melanjutkan batas bayangan secara general.
- c. Bayangan selalu dikaitkan dengan objek yang membuatnya dan perilaku dari objek (contoh : jika seorang membuka lengannya, bayangan akan mencerminkan gerakan dan bentuk orang).
- d. Bentuk bayangan adalah proyeksi dari bentuk objek. Untuk sumber cahaya yang panjang (bukan titik sumber cahaya), proyeksi tidak mungkin perspektif.
- e. Posisi dan kekuatan dari sumber cahaya diketahui dari bayangan.

- f. Ukuran bayangan bergantung pada arah sumber cahaya dan ketinggian objek.

2.2.6. Color Invariant

Meskipun ada kemajuan yang signifikan dalam ketetapan warna komputasi (*color constancy*), algoritma ketetapan warna (*color constancy*) belum menunjukkan tingkat kinerja dan generalitas yang memungkinkan algoritma tersebut diterapkan secara sistematis untuk sebagian besar permukaan objek dalam citra warna [4].

Metode terbaru tentang pengenalan berbasis warna menemukan deskriptor invariant warna (*color invariant*) dari distribusi warna tanpa menggunakan algoritma ketetapan warna (*color constancy*) *full-blown*.

Deskripsi yang disebut “*color invariant*” tidak mencoba untuk memperkirakan penggambaran cahaya dan pantulan permukaan secara langsung. Sebaliknya mereka merupakan fitur yang tidak berubah dengan perubahan cahaya.

Ada berbagai cara berbeda untuk mendapatkan fitur invariant. Satu kemungkinan adalah untuk menormalkan setiap lokasi citra dengan beberapa referensi RGB sehingga diperoleh ruang warna (*color space*) baru yang pikselnya tidak berubah terhadap perubahan cahaya. Jelas bahwa disini faktor penskalaan yang menjelaskan tentang perubahan pencahayaan tidak digunakan. Demikian juga rasio warna antara piksel bersebelahan akan tetap konstan sepanjang perubahan cahaya dan karena itu telah digunakan untuk pengindeksan *color invariant*. Juga memungkinkan untuk memperoleh fitur statistik global dari distribusi warna yang tidak bergantung pada warna cahaya.

Berbagai model *color invariant* telah diusulkan. Cara untuk mengevaluasi dan menganalisisnya bergantung pada kriteria berikut :

- Kesukaran untuk berubah dalam orientasi permukaan (mis. Geometri objek)
- Kesukaran terhadap perubahan intensitas pencahayaan
- Kesukaran terhadap perubahan distribusi daya spektral (SPD) pencahayaan.

a. Model Pemantulan Diskrit

Gambar yang diambil dengan perangkat seperti kamera citra digital terdiri dari respon sensor (dalam kasus permukaan Lambertian) yang dapat dirumuskan dengan [4] :

$$R_k = \sigma \int_{\omega} E(\lambda) S(\lambda) Q_k(\lambda) d\lambda ,$$

$$k = R, G, B \quad (2.1)$$

Misalkan kepekaan sensor kamera warna adalah *narrow-band* sehingga respon spektralnya dapat diperkirakan dengan fungsi Dirac Delta $Q_k(\lambda) = q_k \delta(\lambda - \lambda_k)$, maka persamaan 2.1 menjadi

$$R_k = \sigma E(\lambda_k) S(\lambda_k) q_k \quad (2.2)$$

istilah *shading* σ ditentukan oleh arah sumber cahaya dan permukaan normal. *Shading* mengikuti perubahan posisi relatif pada sumber cahayanya dengan merujuk pada *shading* permukaan. Dengan menganggap perangkat pencitraan trikromatik dimana respon tiga sensor pada satu piksel dilambangkan (R_i, G_i, B_i) , sehingga perubahan pada *shading* dapat dirumuskan dengan :

$$\begin{pmatrix} R_i \\ G_i \\ B_i \end{pmatrix} = \begin{pmatrix} \rho_i R_i \\ \rho_i G_i \\ \rho_i B_i \end{pmatrix} \quad (2.3)$$

dengan ρ_i adalah skalar sederhana yang merepresentasikan perubahan *shading*. Perhatikan bahwa skalar ρ_i memiliki subskrip i yang menunjukkan bahwa semua piksel dapat memiliki faktor kecerahan (*brightness*) masing-masing. Antara lain, perubahan kecerahan (*brightness change*), atau geometri penyorotan (*lighting geometry*), yang merupakan fenomena lokal.

Pandang perubahan warna pencahayaan (dengan asumsi geometri penyorotan dipertahankan tetap). Matriks diagonal dapat secara akurat menggambarkan perubahan warna diberbagai pencahayaan (*illuminations*) untuk sensor *narrow-band*. Dari persamaan 2.1, perubahan pencahayaan sangat mudah dimodelkan : jika (R, G, B) menotasikan respon kamera dari suatu objek dibawah satu warna cahaya, $(\alpha R, \beta G, \gamma B)$ adalah respon yang diinduksi oleh objek yang sama dilihat dari bawah pencahayaan berbeda. *Channel* warna secara sederhana dapat diskalakan dengan matriks diagonal :

$$\begin{pmatrix} R_i \\ G_i \\ B_i \end{pmatrix} = \begin{pmatrix} \alpha & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & \gamma \end{pmatrix} \begin{pmatrix} R_i \\ G_i \\ B_i \end{pmatrix} \quad (2.4)$$

Selanjutnya, saat persamaan 2.4 bukan model yang baik dimungkinkan untuk melakukan transformasi linier RGB untuk memastikan model-model tersebut bertahan.

Semua model diatas bergantung pada kepemilikan linearitas dari proses pencitraan dan pada citra yang dikalibrasi dimana nilai piksel diasumsikan memiliki hubungan linier dengan intensitas aktual dalam *scene*. Sebenarnya, fungsi daya yang diterapkan pada respon sensor mentah sebagai bagian

dari proses *coding* dan fungsi ini dapat bergantung pada perangkat bahkan gambar. R_i, G_i dan B_i dinaikkan ke eksponen tetap yang dapat disebut sebagai *gamma* dan dinotasikan γ :

$$\begin{pmatrix} R_i \\ G_i \\ B_i \end{pmatrix} = \begin{pmatrix} R_i^\gamma \\ G_i^\gamma \\ B_i^\gamma \end{pmatrix} \quad (2.5)$$

Menggabungkan persamaan 2.3 , 2.4 dan 2.5 :

$$\begin{pmatrix} R_i \\ G_i \\ B_i \end{pmatrix} = \begin{pmatrix} \alpha' \rho_i' R_i^\gamma \\ \beta' \rho_i' G_i^\gamma \\ \gamma' \rho_i' B_i^\gamma \end{pmatrix} \quad (2.6)$$

dengan γ dimasukkan kedalam skalar sehingga $\alpha' \rightarrow \alpha^\gamma, \beta' \rightarrow \beta^\gamma, \gamma' \rightarrow \gamma^\gamma$.

Persamaan 2.6 menyatakan bahwa nilai RGB berubah saat satu atau lebih geometri penyinaran (*lighting geometry*), warna pencahayaan (*illumination color*) , dan gamma berubah.

b. Normalisasi Warna

Ruang kromasitas representasi kromatisitas general adalah untuk membagi RGB dengan masing-masing intensitas $R + G + B$:

$$(r, g, b) = \left(\frac{R}{R+G+B}, \frac{G}{R+G+B}, \frac{B}{R+G+B} \right) \quad (2.7)$$

Hanya dengan mengisi R, G, B dari persamaan 2.6, dapat dengan mudah dilihat bahwa dengan memfaktorkan intensitas, ruang kromatisitas ini tidak tergantung pada intensitas dan karena hal tersebut ruang kromatisitas juga tidak tergantung pada geometri pencitraan (posisi objek dan sumber cahaya) dan efek *shading*.

Normalisasi Gray-World Normalisasi *Gray-World* membagi tiap piksel dengan rata-rata dari citra :

$$\left(\frac{R}{\text{mean}(R)}, \frac{G}{\text{mean}(G)}, \frac{B}{\text{mean}(B)} \right) \quad (2.8)$$

$\text{mean}(R)$ menotasikan rata-rata nilai piksel merah untuk citra , $\text{mean}(G)$ rata-rata nilai piksel hijau untuk citra , dan $\text{mean}(B)$ rata-rata nilai piksel biru untuk citra.

Sedangkan untuk normalisasi *gray-world*, faktor *scalling* yang menggambarkan perubahan pencahayaan (*illumination change*) sesuai dengan model diagonal dihapuskan. Dengan demikian citra normalisasi *gray-world* tidak tergantung pada warna cahaya insiden.

Normalisasi *MaxRGB* normalisasi *MaxRGB* menormalkan citra secara global dengan piksel referensi. Namun, disini potongan kecil dari referensi dipilih menjadi yang paling terang pada *scene*

$$\left(\frac{R}{\text{max}(R)}, \frac{G}{\text{max}(G)}, \frac{B}{\text{max}(B)} \right) \quad (2.9)$$

Citra yang ternormalisasi sesuai dengan persamaan 2.9 invarian terhadap pencahayaan *scene* (*scene illumination*).

Normalisasi komprehensif independensi pencahayaan (*illumination independency*) dapat ditemukan dengan menerapkan normalisasi *gray-world*. Di sisi lain, transformasi kromasitas menyajikan variasi intensitas dan karena itu geometri *scene*. Finlayson dkk. mengusulkan bahwa dengan menerapkan secara rekursif setiap normalisasi secara bergilir, sebuah prosedur *normalisasi citra warna komprehensif*, memungkinkan untuk mendapatkan representasi yang merangkum kedua jenis invariant [9].

Normalisasi komprehensif didefinisikan sebagai :

1. Inisialisasi $I_0 = I$
2. Langkah iterasi $I_{i+1} = G(C(I_i))$
3. Kondisi terminasi $I_{i+1} = I_i$

dengan I menotasikan sebuah citra, $G()$ menotasikan normalisasi *gray-world* dan persamaan 2.8 , dan $C()$ menotasikan normalisasi kromasitas dari persamaan 2.7. Dengan kata lain, normalisasi kromasitas dan normalisasi *gray-world* diterapkan secara berurutan dan berulang kali pada citra sampai citra yang dihasilkan konvergen pada suatu titik tetap.

Finlayson membuktikan bahwa proses iteratif selalu konvergen ke titik tetap yang unik. Jika I dan I' menotasikan citra pada *scene* yang sama dimana hanya kondisi penyorotan yang berubah maka $CN(I) = CN(I')$ dimana $CN()$ menotasikan prosedur normalisasi komprehensif. Metode ini telah terbukti mengungguli *gray-world* individu atau fungsi normalisasi.

Finlayson dkk. merumuskan skema normalisasi lain yang disebut prosedur normalisasi komprehensif non-iteratif, dimana iterasi dihindari dengan menggunakan logaritma citra RGB daripada RGB itu sendiri [10]. Persamaan 2.6 memperjelas bahwa pada ruang RGB, ketika kondisi penyorotan berubah, efek pada piksel multiplikatif. Logaritma mengubah perkalian menjadi penambahan sehingga model 2.6 menjadi :

$$\begin{pmatrix} \log(R_i) \\ \log(G_i) \\ \log(B_i) \end{pmatrix} = \begin{pmatrix} \alpha'' + \rho_i'' + \gamma \log(R_i) \\ \beta'' + \rho_i'' + \gamma \log(G_i) \\ \gamma'' + \rho_i'' + \gamma \log(B_i) \end{pmatrix} \quad (2.10)$$

dengan $\alpha'' = \log \alpha'$, $\beta'' = \log \beta'$, $\gamma'' = \log \gamma'$ dan $\rho_i'' = \log \rho_i'$.

Dengan struktur model ini, ketergantungan pada geometri penyorotan (*lighting geometry*) dapat dengan mudah dihapus dengan mengurangi respon log rata-rata pada setiap

piksel, efek warna pencahayaan (*illumination*) dapat dihilangkan dengan cara serupa. Namun, vektor semua respon log R (atau G, atau B) di seluruh citra harus dioperasikan. Dengan cara mengurangi nilai rata-rata log R dari semua nilai piksel log R dan mengurangi nilai rata-rata log G dan rata-rata log B dari semua log G dan semua nilai log B.

Selain itu, Finlayson juga mengatasi masalah pembentukan citra non-linier untuk menghapus ketergantungan warna citra pada gamma, dengan menggunakan *second order statistic*. Singkatnya, dalam ruang log, fungsi daya (gamma) menjadi perkalian dengan skalar : x^γ yang dipetakan ke $\gamma \log x$. Dengan membagi nilai yang dihasilkan dari masing-masing *channel* dari proses normalisasi diatas dengan masing-masing pembagian standar , gamma akan dihilangkan.

c. Rasio Warna

Pendekatan rasio warna biasanya digunakan pada pengindeksan citra warna, yang mana pengindeksan dilakukan dengan menggunakan rasio warna, daripada nilai-nilai langsung dari warna. Dengan memeriksa bagaimana respon warna pada suatu titik bervariasi dan respon tetangganya, rasio warna menekankan pada distribusi batas perbahan warna, tepi , dll, yang merupakan sifat intrinsik citra, dengan demikian sebagian besar tidak terkait dengan faktor eksternal , seperti perubahan pencahayaan dan perubahan geometri citra.

Funt dan Finlayson menyarankan penggunaan rasio warna dengan *Color Constant Color Indexing* (CCCI) [11]

$$\left(\frac{p_1^1}{p_1^2}, \frac{p_2^1}{p_2^2}, \frac{p_3^1}{p_3^2} \right) \quad (2.11)$$

Untuk pengenalan objek, dimana p_i^j menotasikan nilai dari piksel ke i *channel* pada lokasi j . Mudah dilihat dari model diagonal perubahan pencahayaan (*illumination change*) pada persamaan 2.4 untuk melihat bahwa rasio warna tetap konstan di sepanjang perubahan cahaya.

Gevers dan Smeulders memperluas teknik CCCI untuk memperhitungkan tidak hanya ntuk berbagai warna pencahayaan (*illumination color*) tetapi juga untuk efek *shading*. Ruang $m_1 m_2 m_3$ dirumuskan dengan [12] :

$$(m_1, m_2, m_3) = \left(\frac{p_1^1 p_2^2}{p_1^2 p_2^1}, \frac{p_1^1 p_3^2}{p_1^2 p_3^1}, \frac{p_2^1 p_3^2}{p_2^2 p_3^1} \right) \quad (2.12)$$

p_i^j menotasikan nilai dari piksel ke i *channel* pada lokasi j .

Adjeroh dan Lee mengusulkan fitur invariant berbasis rasio warna lain dengan mengamati bagaimana respon warna pada suatu titik bervariasi dengan respon pada tetangganya. Mereka mengambil rasio variasi ini dengan respon warna aktual dengan rumus berikut [12]:

$$\Phi = \frac{p_{x,y} - \frac{1}{m} \sum_{i=1}^m (p_{x,y} - p_{x,y}^i)}{p_{x,y}} \quad (2.13)$$

$p_{x,y}$ menotasikan respon warna pada titik (x, y) pada citra, dan $p_{x,y}^i$ menotasikan respon ke i titik tetangga (x, y) . Rasio dilakukan di setiap *channel* secara terpisah. Rasio yang dihasilkan, meskipun perubahan pencahayaan (*illumination*) tidak berubah, pada dasarnya mengkodekan informasi tentang konteks spasial pada skala kecil dalam citra.

d. Statistik Invarian

Dengan mempelajari distribusi warna, Finlayson dkk. memberikan fitur statistik yang cocok untuk pengindeksan yang kuat, yang disebut sudut warna. Jika menotasikan $v_i = (p_{i1}, p_{i2}, \dots, p_{iN})$, $i = 1, 2, 3$, sebagai vektor dari respon tiap *channel*, maka sudut warna dari sebuah citra dapat dirumuskan [13]

$$\phi_{ij} = \cos^{-1}\left(\frac{v_i v_j}{|v_i| |v_j|}\right) \quad (2.14)$$

Tiga sudut antara *channel* R, G dan B tidak berubah-ubah terhadap warna sumber cahaya.

Fitur statistik lain invarian dirumuskan oleh Healey dan Slater *moment invariant* [14]. Disini Healey dan Slater mendapatkan fitur invariant dari momen histogram warna. Dimulai dengan pengamatan bahwa histogram warna *scene* yang sama dibawah dua lampu dihubungkan oleh transformasi *affine*, matriks transformasi L ditemukan (oleh penguraian Cholesky dari matriks kovarian distribusi warna). Dengan menggunakan transformasi L, respon warna kemudian dapat diubah menjadi representasi dimana nilai eigen dari matriks momen terpusat tidak berubah terhadap perubahan pencahayaan (*illumination*).

e. *Hue* sebagai Invarian

Hue dapat dihitung dengan menggunakan salah satu ruang warna : HSV, HLS, dan HIS. Pada dasarnya, masing-masing kode ruang RGB berkorelasi dengan tiga hal : *hue*, *saturation*, dan *brightness*. *Brightness* berkorelasi dengan besarnya : putih lebih terang daripada abu-abu. *Saturation*

mengukur kemurnian warna : merah terang lebih desaturasi daripada merah jenuh.

HSV, HLS, dan HIS merumuskan *hue* dengan cara yang sama yaitu :

$$H = \cos^{-1} \frac{0.5[(R-G)+(R-B)]}{\sqrt{(R-G)(R-G)+(R-B)(G-B)}} \quad (2.15)$$

Terlihat bahwa *hue* menghapus ketergantungan geometri penyorotan (*lighting geometry*). Artinya, *hue* tidak berubah dengan perubahan *brightness*.

Finlayson dkk. merumuskan bahwa \log *hue* invarian untuk *brightness* dan gamma [15] :

$$H = \tan^{-1} \frac{\log(R)-\log(G)}{\log(R)+\log(G)-2\log(B)} \quad (2.16)$$

Menerapkan transformasi \log ke nilai RGB memindahkan istilah γ dari eksponen ke skalar multiplikatif, kemudian rasio menghapusnya dalam \log *hue*.

2.2.7. Entropi

Teori informasi yang memberikan landasan teoritis untuk mengukur konten informasi, atau ketidakpastian, dari variabel acak yang direpresentasikan sebagai distribusi [16]. Diberikan X variabel acak diskrit dengan alfabet χ dan probabilitas fungsi massa $p(x), x \in \chi$. Entropi *Shannon* dari X didefinisikan sebagai berikut

$$H(X) = -\sum_{x \in \chi} p(x) \log p(x) \quad (2.17)$$

Dengan $p(x) \in [0.0, 1.0]$, $\sum_{x \in \chi} p(x) = 1.0$, dan $-\log p(x)$ melambangkan informasi yang terkait dengan kemunculan tunggal dari x . Satuan dari informasi disebut dengan *bit*.

Umumnya, logaritma diambil pada *base* 2 atau *e*. Untuk kontinuitas probabilitas nol tidak membantu pada entropi, *i.e.*, $0 \log 0 = 0$. Sebagai ukuran rata-rata ketidakpastian dalam X , entropi selalu bernilai positif atau tidak negatif, dan menunjukkan jumlah *bit* rata-rata yang diperlukan untuk mendeskripsikan variabel acak. Semakin tinggi entropi, semakin banyak informasi dari sebuah variabel. Sifat penting dari entropi adalah bahwa $H(x)$ adalah fungsi cekung dan mencapai nilai maksimum $\log|\chi|$ jika dan hanya jika $p(x)$ sama untuk semua x , yaitu ketika distribusi probabilitas bernilai seragam. Kunci dari menerapkan konsep entropi untuk masalah visualisasi terletak pada bagaimana cara menentukan secara acak variabel X dan fungsi probabilitas $p(x)$. Dalam kebanyakan kasus, fungsi probabilitas ini bisa didefinisikan secara heuristik untuk memenuhi kebutuhan aplikasi individual.

2.2.8. Root Mean Square Error (RMSE)

Untuk melihat apakah hasil dari metode dalam penghapusan bayangan digunakan perhitungan RMSE. Sebelum menghitung RMSE perlu diketahui terlebih dahulu tentang *Mean Square Root* (MSE). MSE merupakan ukuran yang digunakan untuk menilai baik buruknya sebuah metode untuk melakukan rekonstruksi atau restorasi citra relatif terhadap citra aslinya [17]. Perhitungan nilai MSE dapat dilakukan dengan persamaan (2.8).

$$MSE = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N [f_1(x, y) - f_2(x, y)]^2 \quad (2.18)$$

dengan M dan N adalah jumlah baris dan kolom pada citra *input* serta $f_1(x, y)$ dan $f_2(x, y)$ adalah citra asli dan citra *output* yang sudah diproses. Semakin kecil nilai dari MSE, menunjukkan bahwa hasil pemrosesan citra semakin bagus.

Untuk perhitungan nilai RMSE (*Root Mean Square Root*) dapat dilakukan dengan persamaan (2.19)

$$RMSE = \sqrt{\frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N [f_1(x, y) - f_2(x, y)]^2} \quad (2.19)$$

sama seperti pada MSE dengan M dan N adalah jumlah baris dan kolom pada citra *input* serta $f_1(x, y)$ dan $f_2(x, y)$ adalah citra asli dan citra *output* yang sudah diproses. Semakin kecil nilai dari RMSE, juga menunjukkan bahwa hasil pemrosesan citra semakin bagus.

BAB III

METODE PENELITIAN

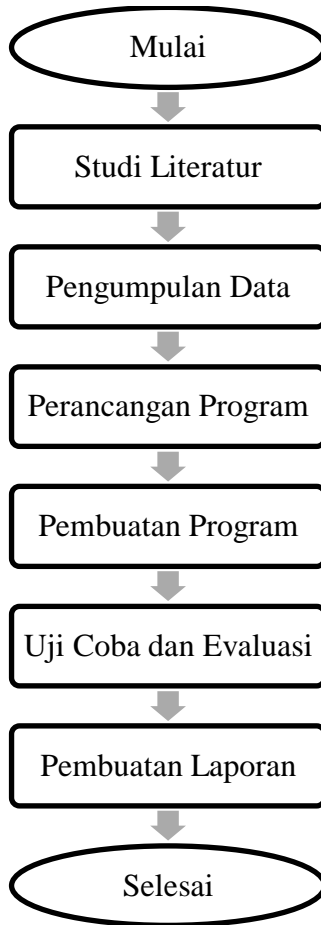
Bab ini menguraikan objek dan data yang digunakan dalam Tugas Akhir penghapusan bayangan dengan menggunakan metode *Color Invariant* pada citra digital serta langkah-langkah sistematis yang dilakukan dalam proses pengerjaan Tugas Akhir ini.

3.1. Data Penelitian

Objek penelitian dalam Tugas Akhir ini adalah citra dengan bayangan yang didapat dari pengambilan data citra sendiri dan *github* yaitu ISTD Dataset dan USF Dataset. Bayangan yang akan dilakukan proses penghapusan adalah *cast shadow* yaitu area yang diproyeksikan oleh objek yang searah dengan cahaya langsung.

3.2. Tahapan Penelitian

Kegiatan penelitian dalam Tugas Akhir ini melalui tahapan-tahapan studi literatur, pengumpulan data, perancangan program, pembuatan program, uji coba dan evaluasi, dan pembuatan laporan, seperti yang ditunjukkan pada Gambar 3.1



Gambar 3.1 Langkah-langkah penelitian

Berikut merupakan penjelasan lebih lanjut mengenai langkah-langkah penelitian penghapusan bayangan pada citra digital :

a. Studi Literatur

Sebelum membuat program untuk menghapus bayangan, akan dilakukan studi literatur terlebih dahulu untuk mengetahui apa saja yang perlu dilakukan untuk membuat program tersebut. Studi literatur yang dilakukan yaitu meliputi Metode *Color Invariant* dan pendalaman tentang bayangan pada citra. Selama proses penelitian pun juga akan terus dilakukan studi literatur untuk mendapatkan hasil penelitian yang lebih maksimal.

b. Pengumpulan Data

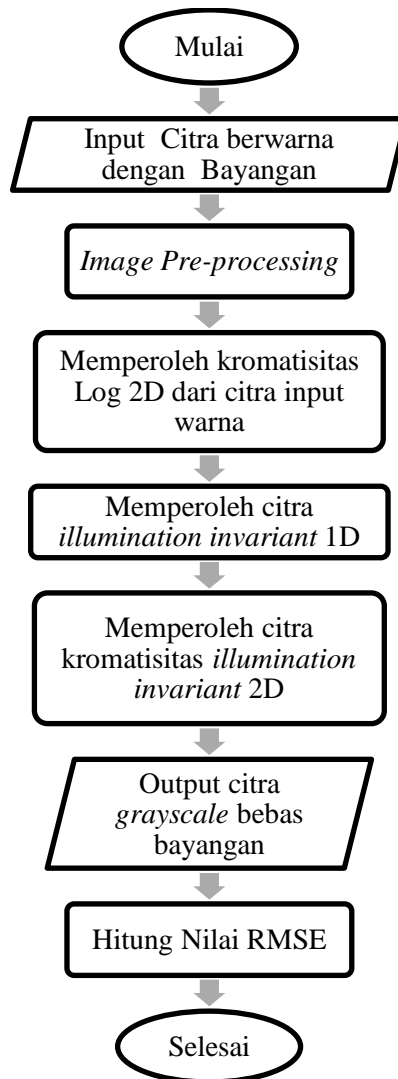
Data adalah hal yang sangat penting dalam pembuatan sebuah penelitian, begitu pula dengan penelitian mengenai penghapusan bayangan ini. Seiring dengan melakukan studi literatur, akan dilakukan pengumpulan data sebagai dasar untuk pembuatan program. Data citra dengan bayangan didapat dari citra yang diambil oleh penulis sendiri dan data citra yang didapat dari website penyedia dataset.

c. Perancangan Program

Dalam tahap ini , dilakukan perancangan terhadap output, prosedur, perangkat lunak dan perangkat keras yang diperlukan dalam pembuatan program. Input dalam program ini adalah citra dengan bayangan, sedangkan output yang akan didapat adalah citra kromatisitas bebas bayangan.

d. Pembuatan Program

Setelah mengumpulkan data dan merancang program, akan mulai dilakukan pembuatan program. Langkah-langkah algoritma penghapusan bayangan pada citra digital ditunjukkan pada Gambar 3.2.



Gambar 3.2 Diagram Alir Penghapusan Bayangan

Penjelasan lebih lanjut mengenai langkah-langkah tersebut adalah sebagai berikut :

1. Input Citra berwarna dengan Bayangan
Pada program ini input program berupa citra berwarna dengan bayangan.
2. *Image Pre-processing*
Input citra dengan bayangan akan diolah terlebih dahulu agar sesuai dengan kebutuhan program yang akan dibuat.
3. Memperoleh kromatisitas Log 2D dari citra input warna
 - Pertama normalkan channels R, G dan B, dari suatu input citra RGB
 - Kedua cari log dari masing – masing channels.
 - Ketiga proyeksikan masing – masing log dari channels ke 2D.
4. Memperoleh citra *illumination invariant* 1D
Setelah mendapat nilai Log kromatisitas dua dimensi , selanjutnya akan diolah dengan metode minimasi entropi.
5. Memperoleh citra kromatisitas *illumination invariant* 2D
Pada tahap ini akan didapat citra kromatisitas bebas bayangan untuk tampilan gambar output lebih baik.
6. Output citra *grayscale* bebas bayangan
Setelah memperoleh citra kromatisitas bebas bayangan. Citra tersebut akan diubah kedalam *grayscale* untuk mempermudah perhitungan RMSE.
7. Hitung Nilai RMSE
Setelah didapatkan *output* citra *grayscale* bebas bayangan, citra tersebut dibandingkan dengan citra asli tanpa bayangan dengan mempertimbangkan citra *ground-truth* dan dihitung nilai MSE untuk mendapatkan nilai RMSE. Pada tugas akhir ini terdapat tiga RMSE yaitu RMSE pada daerah

bayangan pada citra RMSE daerah bukan bayangan dan RMSE Total.

e. Uji Coba dan Evaluasi

Setelah program reduksi noise hujan ini selesai dibuat, akan dilakukan uji coba program apakah program ini sudah valid atau belum dan apakah pengguna dapat menggunakan program ini dengan baik atau belum. Disamping uji coba, akan dilakukan juga evaluasi pada program agar program lebih maksimal.

f. Pembuatan Laporan

Di akhir penelitian ini, akan dilakukan pembuatan laporan. Laporan dari program penghilang noise hujan ini akan dibuat setelah program lolos dari uji coba dan telah menjawab evaluasi yang didapat.

BAB IV

PERANCANGAN DAN IMPLEMENTASI PROGRAM

Bab ini menjelaskan lebih lanjut proses perancangan dan implementasi program penghapusan bayangan pada citra digital dengan menggunakan metode *Color Invariant*.

4.1. Perancangan Program

Langkah – langkah yang dilakukan untuk penghapusan bayangan pada citra digital dengan menggunakan metode *Color Invariant* adalah dengan tahap penormalan pada tiap *channel* RGB dan menerapkan metode Minimasi Entropi.

Gambar 4.1 merupakan contoh citra Bayangan manusia dengan dimensi 1600 x 1200 piksel. Pada pembahasan perancangan program ini, akan diambil citra *input* yaitu citra bayangan manusia itu sendiri yang selanjutnya akan di *resize* menjadi 200 x 200 piksel seperti pada Gambar 4.2.



Gambar 4.1 Bayangan Manusia 1600x 1200 piksel



Gambar 4.2 Citra input 200 x 200 piksel

Proses diatas merupakan *pre-Processing* agar pengolahan gambar bisa lebih cepat. Gambar tersebut selanjutnya akan diproses lebih lanjut seperti pada diagram 3.2. Dengan rincian sebagai berikut :

a. Memperoleh kromatisitas Log 2D dari citra input warna

1. Langkah Pertama yaitu menormalkan *channels* R, G dan B, dari gambar inputan. Langkah ini dilakukan untuk mengurangi efek pencahayaan pada gambar sehingga dapat dibedakan bagian dari bayangan dan bukan bagian dari bayangan.

Perhitungan untuk menormalkan tiap *channels* RGB dapat dilakukan sesuai dengan persamaan (4.1) berikut [18] :

$$\begin{aligned}
 R' &= \frac{R}{R + G + B} \\
 B' &= \frac{B}{R + G + B} \\
 G' &= \frac{G}{R + G + B}
 \end{aligned}
 \tag{4.1}$$

R' , G' , dan B' merupakan *channel* R, G dan B yang telah dinormalkan.

Berikut diambil contoh perhitungan manual pada 5 x 5 piksel citra *input* :

- Matriks citra *input channel Red* (R)

$$R = \begin{bmatrix} 54 & 72 & 49 & 60 & 67 \\ 79 & 81 & 40 & 74 & 87 \\ 110 & 80 & 49 & 73 & 77 \\ 72 & 47 & 63 & 70 & 56 \\ 52 & 46 & 81 & 66 & 41 \end{bmatrix}$$

- Matriks citra *input channel Green* (G)

$$G = \begin{bmatrix} 70 & 105 & 61 & 69 & 85 \\ 83 & 94 & 49 & 88 & 97 \\ 110 & 79 & 55 & 93 & 97 \\ 93 & 60 & 83 & 103 & 82 \\ 75 & 57 & 97 & 103 & 70 \end{bmatrix}$$

- Matriks citra *input channel Blue* (B)

$$B = \begin{bmatrix} 23 & 17 & 21 & 29 & 30 \\ 42 & 40 & 22 & 46 & 51 \\ 56 & 43 & 28 & 26 & 33 \\ 28 & 24 & 32 & 20 & 19 \\ 21 & 27 & 39 & 20 & 15 \end{bmatrix}$$

Matriks – matriks diatas akan diubah menjadi presisi double dengan fungsi `im2double()` pada Matlab. Sehingga matriks menjadi :

- Matriks citra *input channel Red* (R) setelah dikenakan `im2double()`

$$R = \begin{bmatrix} 0.2118 & 0.2824 & 0.1922 & 0.2353 & 0.2627 \\ 0.3098 & 0.3176 & 0.1569 & 0.2902 & 0.3412 \\ 0.4314 & 0.3137 & 0.1922 & 0.2863 & 0.3020 \\ 0.2824 & 0.1843 & 0.2471 & 0.2745 & 0.2196 \\ 0.2039 & 0.1804 & 0.3176 & 0.2588 & 0.1608 \end{bmatrix}$$

- Matriks citra *input channel Green* (G) setelah dikenakan `im2double()`

$$G = \begin{bmatrix} 0.2745 & 0.4118 & 0.2392 & 0.2706 & 0.3333 \\ 0.3255 & 0.3686 & 0.1922 & 0.3451 & 0.3804 \\ 0.4314 & 0.3098 & 0.2157 & 0.3647 & 0.3804 \\ 0.3647 & 0.2353 & 0.3255 & 0.4039 & 0.3216 \\ 0.2941 & 0.2235 & 0.3804 & 0.4039 & 0.2745 \end{bmatrix}$$

- Matriks citra *input channel Blue* (B) setelah dikenakan `im2double()`

$$B = \begin{bmatrix} 0.0902 & 0.0667 & 0.0824 & 0.1137 & 0.1176 \\ 0.1647 & 0.1569 & 0.0863 & 0.1804 & 0.2000 \\ 0.2196 & 0.1686 & 0.1098 & 0.1020 & 0.1294 \\ 0.1098 & 0.0941 & 0.1255 & 0.0784 & 0.0745 \\ 0.0824 & 0.1059 & 0.1529 & 0.0784 & 0.0588 \end{bmatrix}$$

- Matriks hasil penjumlahan R, G, dan B

$$S = \begin{bmatrix} 0.5765 & 0.7608 & 0.5137 & 0.6196 & 0.7137 \\ 0.8000 & 0.8431 & 0.4353 & 0.8157 & 0.9216 \\ 1.0824 & 0.7922 & 0.5176 & 0.7529 & 0.8118 \\ 0.7569 & 0.5137 & 0.6980 & 0.7569 & 0.6157 \\ 0.5804 & 0.5098 & 0.8510 & 0.7412 & 0.4941 \end{bmatrix}$$

Setelahnya akan dihitung nilai normal dari tiap *channel* diatas menggunakan persamaan (4.1) sehingga didapat :

- $R' = \begin{bmatrix} 0.3673 & 0.3711 & 0.3740 & 0.3797 & 0.3681 \\ 0.3873 & 0.3767 & 0.3604 & 0.3558 & 0.3702 \\ 0.3986 & 0.3960 & 0.3712 & 0.3802 & 0.3720 \\ 0.3731 & 0.3588 & 0.3539 & 0.3627 & 0.3567 \\ 0.3514 & 0.3538 & 0.3733 & 0.3492 & 0.3254 \end{bmatrix}$

- $G' = \begin{bmatrix} 0.4762 & 0.5412 & 0.4656 & 0.4367 & 0.4670 \\ 0.4069 & 0.4372 & 0.4414 & 0.4231 & 0.4128 \\ 0.3986 & 0.3911 & 0.4167 & 0.4844 & 0.4686 \\ 0.4819 & 0.4580 & 0.4663 & 0.5337 & 0.5223 \\ 0.5068 & 0.4385 & 0.4470 & 0.5450 & 0.5556 \end{bmatrix}$

$$\bullet \quad B' = \begin{bmatrix} 0.1565 & 0.0876 & 0.1603 & 0.1835 & 0.1648 \\ 0.2059 & 0.1860 & 0.1982 & 0.2212 & 0.2170 \\ 0.2029 & 0.2129 & 0.2121 & 0.1354 & 0.1594 \\ 0.1451 & 0.1832 & 0.1798 & 0.1036 & 0.1210 \\ 0.1419 & 0.2077 & 0.1797 & 0.1058 & 0.1190 \end{bmatrix}$$

2. Langkah kedua untuk memperoleh kromatisitas log 2D yaitu dengan mencari log dari masing – masing channels. Dengan persamaan (4.2) sebagai berikut :

$$\begin{aligned} LR' &= \log_{10}(R'); \\ LG' &= \log_{10}(G'); \\ LB' &= \log_{10}(B'); \end{aligned} \tag{4.2}$$

Berikutnya akan dicari nilai log() dari R' , G' dan B' dengan hasil sebagai berikut :

$$\bullet \quad LR' = \begin{bmatrix} -0.4349 & -0.4305 & -0.4271 & -0.4205 & -0.4340 \\ -0.4120 & -0.4240 & -0.4433 & -0.4488 & -0.4315 \\ -0.3995 & -0.4023 & -0.4304 & -0.4200 & -0.4295 \\ -0.420 & -0.4452 & -0.4511 & -0.4405 & -0.4477 \\ -0.4543 & -0.4512 & -0.4280 & -0.4569 & -0.4876 \end{bmatrix}$$

$$\bullet \quad LG' = \begin{bmatrix} -0.3222 & -0.2666 & -0.3319 & -0.3598 & -0.3307 \\ -0.3906 & -0.3593 & -0.3551 & -0.3736 & -0.3843 \\ -0.3995 & -0.4077 & -0.3802 & -0.3148 & -0.3292 \\ -0.3171 & -0.3391 & -0.3313 & -0.2727 & -0.2821 \\ -0.2952 & -0.3581 & -0.3497 & -0.2636 & -0.2553 \end{bmatrix}$$

$$\bullet \quad LB' = \begin{bmatrix} -0.8056 & -1.0574 & -0.7951 & -0.7363 & -0.7830 \\ -0.6864 & -0.7304 & -0.7029 & -0.6553 & -0.6635 \\ -0.6927 & -0.6719 & -0.6734 & -0.8683 & -0.7975 \\ -0.8384 & -0.7371 & -0.7453 & -0.9845 & -0.9171 \\ -0.8480 & -0.6826 & -0.7454 & -0.9754 & -0.9243 \end{bmatrix}$$

3. Karena LR' , LG' dan LB' merupakan vektor 3D bukan 2D, maka langkah selanjutnya adalah memproyeksikan masing – masing log dari channels ke 2D. Berikut rumus untuk mendapatkan x_1 dan x_2 yang merupakan hasil proyeksi LR' , LG' dan LB' ke bidang 2D [19] :

$$x_1 = LR' * \frac{1}{\sqrt{2}} - LG' * \frac{1}{\sqrt{2}} \quad (4.3)$$

$$x_2 = LR' * \frac{1}{\sqrt{6}} + LG' * \frac{1}{\sqrt{6}} - LB' * \frac{2}{\sqrt{6}}$$

Dengan hasil sebagai berikut :

$$x_1 = \begin{bmatrix} -0.0797 & -0.1159 & -0.0673 & -0.0429 & -0.0731 \\ -0.0152 & -0.0457 & -0.0623 & -0.0532 & -0.0334 \\ 0 & 0.0039 & -0.0355 & -0.0744 & -0.0709 \\ -0.0786 & -0.0750 & -0.0847 & -0.1186 & -0.1171 \\ -0.1125 & -0.0658 & -0.0554 & -0.1367 & -0.1643 \end{bmatrix}$$

$$x_2 = \begin{bmatrix} 0.3487 & 0.5787 & 0.3393 & 0.2826 & 0.3271 \\ 0.2328 & 0.2766 & 0.2480 & 0.1993 & 0.2087 \\ 0.2394 & 0.2179 & 0.2189 & 0.4090 & 0.3414 \\ 0.3803 & 0.2816 & 0.2891 & 0.5127 & 0.4509 \\ 0.3865 & 0.2269 & 0.2911 & 0.5023 & 0.4514 \end{bmatrix}$$

b. Memperoleh citra *illumination invariant* 1D

Untuk mendapatkan citra 1D ,citra yang telah diolah sebelumnya akan dikenakan rumus sebagai berikut :

$$I_{inv} = x_1 \cos(\theta) + x_2 \sin(\theta) \quad (4.4)$$

Rumus tersebut akan menghasilkan citra *illumination invariant* 1D (*grayscale*) dimana efek gelap dari bayangan sudah berkurang. Untuk memperoleh nilai θ yang sesuai dapat digunakan metode yaitu *Entropy Minimization*.

Entropy Minimization digunakan untuk memperoleh nilai θ yang sesuai agar proses penghapusan atau reduksi bayangan dapat optimal. Citra yang telah dinormalkan dan diproyeksikan selanjutnya dikenakan *entropy minimization* dengan rumus sebagai berikut :

$$H(X) = - \sum_{x \in \chi} p(x) \log p(x) \quad (4.5)$$

Selanjutnya untuk mendapat kan nilai θ , dipilih dengan cara mencari nilai minimum dari $H(X)$.

$$\theta = \min(H(X)) \quad (4.6)$$

- Langkah untuk mendapat I_{inv} pertama yaitu menentukan *degree* dengan menggunakan fungsi *Invangle* dibawah ini

Algoritma fungsi InvAngle

```

• Inisialisasi nilai epsilon 0.001
• Inisialisasi histbins zeros(64 x 1)
• insialisasi subs = 1
• chA dan chB adalah nilai x1 dan x2

for angle = 0 :0.0175:pi
    inv = 255*(cos(angle)*(chA) -sin(angle)*(chB)

    inv = inv – min(min(inv))

    inv = inv*255/max(max(inv))

    inv = round(inv) (Pembulatan nilai inv)

• inisialisasi u = 1
for I = 0:4:255
    [m n] = size( inv ≥ I dan inv ≤ (i-3))
    histBins(u) = m
    u = u +1 dimana u fungsi jumlahan
end

• Perhitungan Minimasi entropi
    [m n] = size(inv)
    histBins = histBins/(m*n)
    histBins(histBins = 0) = epsilon
    logs = log(histBins)
    res = hisBins x logs

```

```

Eta(subs) = sum(res)*-1
subs = subs + 1
end

deg = find(eta==min(eta))-1

(nilai deg didapat dari nilai minimum eta - 1)

```

- Selanjutnya jalankan fungsi pada *script* dimana $\text{deg} = \text{invangle1}(x1, x2)$, dimana $x1$ dan $x2$ sudah diperoleh dari proses sebelumnya.

Algoritma untuk mendapatkan citra *illumination invariant* 1D

- Cari nilai $\text{deg} = \text{invAngle}(x1, x2)$
- Cari nilai $\text{rad} (\theta) = \text{deg} * (\pi/180)$
- Memperoleh citra 1D *illumination invariant*
 $\text{Inv} = (\cos(\text{rad})*(x1) - \sin(\text{rad})*(x2))$
- Cari $\text{invexp} = \exp(\text{inv})$
- Nilai invexp akan dihitung kembali dengan rumus :
 $\text{invexp} = \text{invexp} - \min(\min(\text{invexp}))$
 $\text{invexp} = \text{invexp} * 255 / \max(\max(\text{invexp}))$

c. Memeroleh citra kromatisitas *illumination invariant* 2D

Sebenarnya pada saat diperoleh citra *illumination invariant* 1D efek gelap dari bayangan sudah tereduksi. Namun untuk tampilan yang lebih baik maka citra 1D diatas akan diproyeksikan ke citra 2D, dengan membagi menjadi dua *clustering* sebagai berikut :

1. Mencari rata-rata dari x_1 atau x_2

$$x = (\max(x_2) + \min(x_2))/2$$
2. Gunakan x untuk membagi nilai x_1 dan ambil nilai tengah dari keduanya.
3. Rekontruksi citra Kromatisitas dengan menggunakan citra *invariant* 1D, sebagai berikut :

Jika $x_2 \geq x$ maka

$$RG = I_{inv} \cos(-\theta) + m_2 \sin(-\theta)$$

$$BG = -I_{inv} \sin(-\theta) + m_2 \cos(-\theta)$$

Jika $x_2 < x$ maka

$$RG = I_{inv} \cos(-\theta) + m_2 \sin(-\theta)$$

$$BG = -I_{inv} \sin(-\theta) + m_2 \cos(-\theta)$$

4. Karena RG dan BG berada pada domain logaritma, sehingga harus didapatkan nilai eksponensial dari RG dan BG untuk menghasilkan citra kromatisitas *Illumination invariant* 2D yang sudah berkurang efek gelap bayangannya.
5. Algoritma dari langkah diatas adalah sebagai berikut :

- Cari rata – rata dari x_1 atau x_2

$$x = (\max(x_2) + \min(x_2))/2$$
- Cari masing –masing nilai dibawah ini :

$$c_1 = \text{nilai dari } x_2 \geq x$$

$$c_2 = \text{nilai dari } x_2 \geq x$$

$$m_1 = \text{median}(c_1)$$

$$m_2 = \text{median}(c_2)$$
- Rekonstruksi citra kromatisitas
 m , n ukuran baris dan kolom dari citra
for $i = 1:m$
 for $j = 1:n$
 if $x_1 \geq x$
 $RG(i, j) = I_{inv}(i, j) \cos(-\theta) + m_1 \sin(-\theta)$
 $BG(i, j) = -I_{inv}(i, j) \sin(-\theta) + m_1 \cos(-\theta)$
 else $\%(x_1 < x)$
 $RG(i, j) = I_{inv}(i, j) \cos(-\theta) + m_2 \sin(-\theta)$
 $BG(i, j) = -I_{inv}(i, j) \sin(-\theta) + m_2 \cos(-\theta)$
 end
 end
end
- Cari nilai eksponen dari RG dan BG

$$\text{resRG} = \exp(RG)$$

$$\text{resBG} = \exp(BG)$$

- Menampilkan citra kromatisitas
 1. Menentukan tiap channel r, g dan b dengan rumus berikut :

$$b = (3 * \text{invexp}) / (\text{resRG} + \text{resBG} + 1)$$

$$b = b - \min(b(:))$$

$$b = b / \max(b(:))$$

$$r = \text{resRG} * b$$

$$r = r - \min(r(:))$$

$$r = r / \max(r(:))$$

$$g = \text{resBG} * b$$

$$g = g - \min(g(:))$$

$$g = g / \max(g(:))$$
 2. Membuat persamaan sebagai berikut

$$\text{Krom}(:, :, 1) = r \text{ (channel red)}$$

$$\text{Krom}(:, :, 2) = g \text{ (channel green)}$$

$$\text{Krom}(:, :, 3) = b/2 \text{ (channel blue)}$$
 3. Menampilkan citra kromatisitas

$$\text{imshow(Krom)}$$

d. Merekonstruksi citra kromatisitas *Illumination invariant* 2D menjadi citra *Grayscale*.

Pada tugas akhir ini citra kromatisitas *Illumination invariant* 2D yang sudah berkurang efek bayangannya akan diubah menjadi citra *grayscale*. Kendala yang muncul adalah citra *illumination invariant* 2D tidak dapat dikembalikan kedalam citra RGB dikarenakan pada tahap sebelumnya terdapat proses memunculkan *illumination invariant* 1D yang berupa citra *grayscale* sehingga beberapa informasi tentang RGB sudah terhapus. Sehingga penulis mencoba cara lain yaitu dengan mengubah citra kromatisitas *illumination invariant* 2D ke citra *grayscale*.

Algoritma

- Buat variabel grayImage ubah ke citra *grayscale*
 $\text{grayImage} = \text{rgb2gray}(\text{Krom})$
- Untuk menyesuaikan pencahayaan pada citra output dilakukan invers dengan langkah berikut :
 $u = 1 - \text{grayImage}$

4.2. Implementasi Program

Program yang telah dirancang sesuai dengan rincian diatas, kemudian diimplementasikan ke dalam bahasa komputer. Pada implementasi program ini digunakan MATLAB R2020a untuk menjalankan rancangan diatas. Proses yang dilakukan saat implementasi program adalah sebagai berikut :

a. Proses input citra

Langkah pertama yang dilakukan pada program ini adalah menginputkan citra dengan bayangan. Citra yang

diinputkan sebaiknya berukuran lebih dari 200 x 200 piksel agar sesuai dengan tahap perancangan yang telah dilakukan. Penginputan citra dapat dilakukan dengan cara berikut :

```
b=imread('Image9.jpg');
```



Gambar 4. 3 Citra *input* 640x 480 piksel

b. Proses *Resize* Citra Input

Setelah citra diinputkan, citra di *resize* dengan cara sebagai berikut :

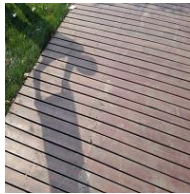
```
b=imresize(b, [200 200]);  
a= im2double(b);
```

Citra input *b* dikenakan *im2double* untuk mengkonversi citra *b* kedalam double dan dihasilkan nilai-nilai keluaran antara 0 sampai 1.

c. Proses menampilkan citra

Citra yang sudah diproses sesuai dengan yang dijabarkan pada perancangan program akan ditampilkan dengan cara sebagai berikut :

```
figure;  
imshow(b);
```



Gambar 4. 4 Citra *resize* 200 x 200 piksel

Proses menentukan tiap *channel* RGB dari sebuah citra yaitu sebagai berikut :

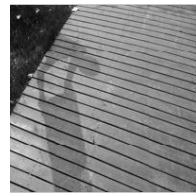
```
r = a(:, :, 1);  
g = a(:, :, 2);  
b = a(:, :, 3);
```



(a)



(b)



(c)

Gambar 4. 5 (a) Citra *input channel Red* , (b) Citra *input channel Green* , (c) Citra *input channel Blue*

- d. Memperoleh kromatisitas Log 2D dari citra input warna

Setelah tiap *channel* RGB dari sebuah citra telah dikenali selanjutnya akan dilakukan proses penormalan tiap *channel* RGB sebagai berikut :

$$\begin{aligned} S &= r+g+b; \\ RN &= r./S; \\ GN &= g./S; \\ BN &= b./S; \end{aligned}$$

RN , GN dan BN merupakan *channel* yang telah dinormalkan. Selanjutnya adalah mendapat nilai log dari tiap *channel* yang telah dinormalkan sebagai berikut :

$$\begin{aligned} LRN &= \log_{10}(RN); \\ LGN &= \log_{10}(GN); \\ LBN &= \log_{10}(BN); \end{aligned}$$

Memproyeksikan LRN, LGN dan LBN kedalam bidang 2D. Sehingga diperoleh nilai x1 dan x2 sebagai berikut :

$$\begin{aligned} x1 &= LRN*1/(\sqrt{2})-LGN*1/(\sqrt{2}); \\ x2 &= LRN*1/(\sqrt{6})+LGN*1/(\sqrt{6})- \\ &\quad LBN*2/(\sqrt{6}); \end{aligned}$$

- e. Memperoleh citra *illumination invariant* 1D

Pada proses ini terjadi proses penghapusan, dimana efek gelap dari bayangan pada sebuah citra akan direduksi. Pada proses ini digunakan *Entropy Minimization* untuk mendapatkan θ yang sesuai sehingga hasil penghapusan bayangan dapat optimal. fungsi `invAngle1` merupakan penerapan rumus *Entropy Minimization*. Berikut lampirannya :

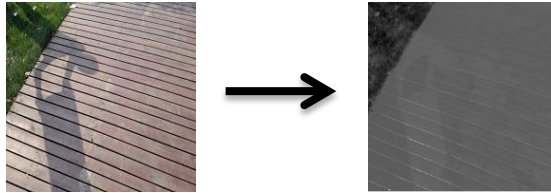
```

function deg=invAngle1(chA, chB)
    epsilon=1e-3;
    histBins=zeros(64,1);
    subs=1;
    for angle=0:0.0175:pi
        inv=255*(cos(angle)*(chA) -
sin(angle)*(chB));
        inv=inv-min(min(inv));
        inv = inv*255/max(max(inv));
        inv=round(inv);
        u=1;
        for i=0:4:255
            [m n]=size(find(inv>=i &
inv<=(i+3)));
            histBins(u)=m;
            u=u+1;
        end
    % Entropy calculation
        [m n]=size(inv);
        histBins = histBins / (m*n);
        histBins(find(histBins ==
0))=epsilon;
        logs = log(histBins);
        res = histBins .* logs;
        eta(subs) = sum(res)*-1;
        subs=subs+1;
    end
    % plot(eta)
    deg = find(eta==min(eta))-1;

```

Berikut merupakan perhitungan rumus setelah didapatkan hasil dari *Entropy Minimization*.

```
deg = invAngle1(x1, x2);
%deg= 20;
rad = deg*(pi/180);
inv = (cos(rad)*(x1) - sin(rad)*(x2));
invexp = exp(inv);
invexp=invexp-min(min(invexp));
invexp=invexp*255/max(max(invexp));
```



Gambar 4. 6 Citra *input illumination invariant 1D*

- f. Memperoleh citra kromatisitas *illumination invariant 2D* bebas bayangan

```
% clustering into 2 clusters

x= (max(x2(:)) + min(x2(:)))/2;
c1=x2(find(x2 >= x));
c2=x2(find(x2 < x));
mc1=median(c1);
mc2=median(c2);
for i=1:200
    for j=1:200
        if(x2(i,j) >=x)
            logresRG(i,j) = inv(i,j)*cos(-
rad) + mc1*sin(-rad);
            logresBG(i,j) = -
inv(i,j)*sin(-rad) + mc1*cos(-rad);
        else
            logresRG(i,j) = inv(i,j)*cos(-
```



```

rad) + mc2*sin(-rad);
        logresBG(i,j) = -
inv(i,j)*sin(-rad) + mc2*cos(-rad);
    end
end
end
resRG = exp(logresRG);
resBG = exp(logresBG);

b =(3*invexp)./ (resRG + resBG +1);
b=b-min(b(:));
b=b/(max(b(:)));

r = resRG.*b;
r=r-min(r(:));
r=r/(max(r(:)));

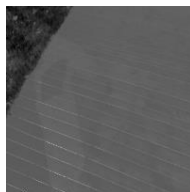
g = resBG.*b;
g=g-min(g(:));
g=g/(max(g(:)));

Krom(:,:,1)=r;
Krom(:,:,2)=g;
Krom(:,:,3)=b/2;
figure('Name','Citra Kromatisitas
','NumberTitle','off');
imshow(Krom);

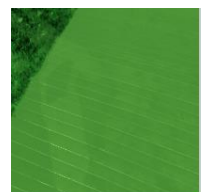
```



(a)



(b)



(c)

Gambar 4. 7 (a) Citra *input* , (b) Citra *iillumination invariant* 1D , (c) Citra Kromatisitas

g. Mengubah citra Kromatisitas menjadi *Grayscale*

```
grayImage = rgb2gray(Krom);
imshow(grayImage);
%u adalah invers dari grayImage untuk
menyesuaikan pencahayaan
u = 1 - grayImage;
figure, imshow(u);
```

Hasil citra kromatisitas diubah menjadi *grayscale* untuk mempermudah perhitungan RMSE. Untuk u invers dari citra *grayscale* tidak diterapkan ke semua data citra yang dipakai hanya digunakan untuk menyesuaikan pencahayaan dari citra *output* agar mirip dengan citra asli *grayscale* tanpa bayangan.

h. Hitung RMSE

RMSE atau *Root Mean Square Error* digunakan untuk membandingkan antara bagian bayangan dan non bayangan dari citra tanpa bayangan dan citra hasil reduksi berdasarkan gambar biner *ground-truth*.

```
IG = getimage(handles.axes15);
Ig1 = im2bw(IG);
IA = getimage(handles.axes7);
Ia = im2double(IA);
IH = getimage(handles.axes8);
Ih = im2double(IH);
totMSENon = 0;
totMSEBy = 0;
totMSE = 0;
[m, n, numberchannel] = size(IA);
for i=1:m
    for j=1:n
        %squareerror = (Ia(i,j)-
Ih(i,j)).^2;
        if Ig1(i,j) == 1
```

```

        totMSEBy = totMSEBy +
        ((Ia(i,j)-Ih(i,j)).^2);
    else
        totMSENon = totMSENon +
        ((Ia(i,j)-Ih(i,j)).^2);
    end
    totMSE = totMSE + ((Ia(i,j)-
    Ih(i,j)).^2);
    end
end
%RMSENon
RMSENon = sqrt((totMSENon)/(m*n));
%RMSEBy
RMSEBy = sqrt((totMSEBy)/(m*n));
%RMSE seluruh
RMSESeluruh = sqrt((totMSE)/(m*n));
%Menampilkan nilai RMSE
set(handles.text19,'String',RMSEBy);
set(handles.text20,'String',RMSENon);
set(handles.text20,'String',RMSETotal);

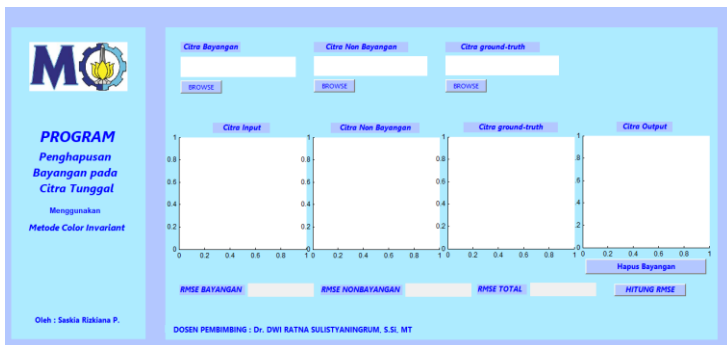
```

IG adalah citra *ground-truth* untuk memastikan IG adalah citra biner maka citra IG dikenakan ke fungsi `im2bw()`. Pada citra *ground-truth* warna hitam bernilai “0” dan warna putih bernilai “1”. IA merupakan citra asli tanpa bayangan yang diubah menjadi *grayscale* dan agar dapat dilakukan proses perhitungan RMSE maka IA dikenakan kedalam fungsi `im2double()`. IH merupakan citra hasil penghapusan yang juga dikenakan ke fungsi `im2double()`. Ukuran atau size dari ketiga citra diatas harus sama disini ukuran nya adalah 200 x 200 piksel. RMSE bayangan menunjukkan nilai error dari bagian bayangan pada citra asli *grayscale* dan citra hasil. Sedangkan RMSE non menunjukkan nilai error dari bagian bukan bayangan. RMSE total adalah RMSE keseluruhan tanpa mempertimbangkan bagian bayangan dan non bayangan.

Disini citra IA atau citra tanpa bayangan dan IH atau citra hasil penghapusan diubah menjadi citra *grayscale* dikarenakan dari metode yang digunakan tidak dapat mengubah citra kromatisitas kembali menjadi citra rgb. Sehingga diambil jalan tengah yaitu mengubah citra IA atau citra asli tanpa bayangan menjadi *grayscale* agar dapat dilakukan perbandingan dan menghasilkan nilai RMSE. Tujuan dari hasil program ini adalah untuk *pre-processing* pengolahan citra yang lebih lanjut, sedangkan untuk diolah lebih lanjut citra yang digunakan bukan merupakan citra berwarna namun citra *grayscale*.

4.3. Tampilan Interface

Interface atau bisa juga disebut antarmuka adalah hal yang penting dalam pembuatan suatu program. Dianggap penting karena *Interface* merupakan hal pertama yang dilihat oleh *User* atau pengguna. *Interface* juga harus bersifat *user compability* atau bisa digunakan dengan baik oleh pengguna. Gambar 4.3 merupakan tampilan *interface* dari program penghapusan bayangan pada citra digital :



Gambar 4. 8 Tampilan interface

Cara kerja dari program ini yang pertama adalah dengan *input* citra bayangan menekan tombol “*Browse*” pada kolom “Citra bayangan”. Setelahnya dengan cara yang sama *input* citra tanpa bayangan dan citra *ground truth*. Selanjutnya setelah semua citra telah diinputkan, proses penghapusan citra bayangan akan berjalan dengan menekan tombol “Hapus bayangan”. Hasil dari penghapusan bayangan akan terlihat pada kotak axes “Citra *output*”. Lalu tekan tombol “Hitung RMSE” untuk menampilkan hasil RMSE bayangan, non bayangan dan total yang merupakan perbandingan dari citra tanpa bayangan *grayscale* dengan citra *output*.

BAB V

UJI COBA DAN ANALISA HASIL

Pada bab ini menjabarkan mengenai data uji coba yang digunakan dan analisa hasil yang didapat dari data uji coba tersebut.

5.1. Data Uji Coba




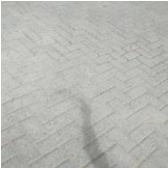



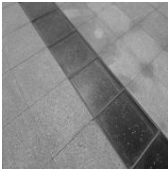

Sebuah program pasti membutuhkan data untuk menjalankannya. Pada program penghapusan bayangan dengan metode *Color Invariant* ini menggunakan data berupa citra berwarna dengan format .jpg dan .png, meliputi :



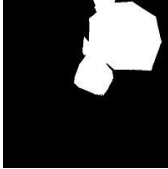

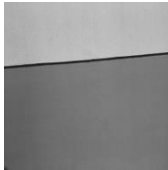

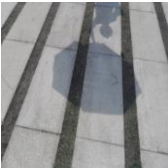



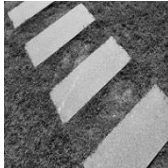

- a. Citra asli dengan bayangan berwarna
- b. Citra asli tanpa bayangan *grayscale*
- c. Citra *ground-truth* bayangan





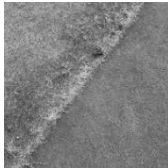


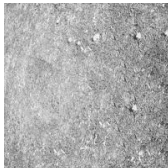




Citra – citra tersebut mengandung bayangan buatan atau disesuaikan waktu pengambilannya agar lebih mudah menghitung perbandingan antara citra yang telah diproses untuk menghapus bayangan, citra asli tanpa bayangan dan citra *ground-truth* bayangan. Cara yang digunakan untuk memperoleh bayangan adalah dengan membuat bayangan sendiri, contohnya adalah dengan memposisikan kamera pada area yang terbebas bayangan lalu memunculkan bayangan menggunakan objek tertentu agar terbentuk bayangan pada area bebas bayangan. Sedangkan untuk citra *ground-truth* diperoleh dari hasil pengeditan dengan *picsart* , dataset serta github. Ukuran dari citra asli tanpa bayangan dan citra *ground-truth* juga disesuaikan dengan citra *output* yang berukuran 200 x 200 piksel. Data citra dengan bayangan selanjutnya akan diproses menggunakan metode yang telah dijelaskan pada bab sebelumnya dengan beberapa variasi kondisi citra yang dijabarkan pada hasil uji coba.













Dari syarat-syarat diatas, didapatkan citra yang digunakan sebagai data uji coba. Citra - citra tersebut ditunjukkan pada Tabel 5.1.













Tabel 5. 1. Data Uji Coba






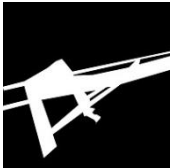



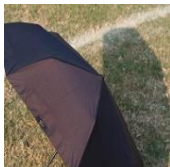


No	Nama citra	Citra Input	Citra non Bayangan grayscale	Citra <i>ground truth</i>
<i>Background citra input cenderung abu-abu dan hitam</i>				
1.	Jalan			
2.	Paving			
3.	Keramik papan			




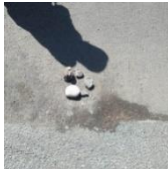

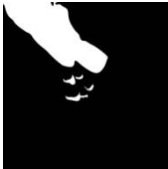






No	Nama citra	Citra Input	Citra non Bayangan grayscale	Citra <i>ground truth</i>
<i>Background citra input cenderung abu-abu dan hitam</i>				
4.	Keramik payung			
5.	Dinding			
6.	Paving rumput			
<i>Background citra input berwarna</i>				
7.	Batu rumput			


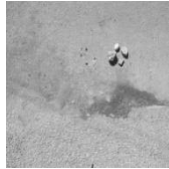

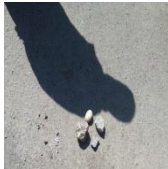


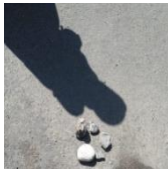





No	Nama citra	Citra Input	Citra non Bayangan grayscale	Citra <i>ground truth</i>
<i>Background citra input berwarna</i>				
8.	Kayu rumput			
9.	Rump ut tanah			
10.	Rump ut			
11.	Jalan rumput			










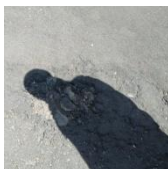

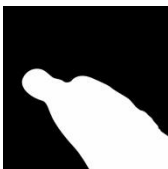
No	Nama citra	Citra Input	Citra non Bayangan <i>grayscale</i>	Citra <i>ground truth</i>
<i>Background citra input berwarna</i>				
12.	Keram ik rumpu t			
<i>Citra bayangan dengan objek</i>				
13.	Kertas dompe t			
14.	Orang dompe t			
15.	Orang balok			



No	Nama citra	Citra Input	Citra non Bayangan grayscale	Citra <i>ground truth</i>
Citra bayangan dengan objek				
16.	Kertas balok			
17.	Kepala balok			
18.	Orang bangk u			
Citra bayangan dengan objek pembentuknya				
19.	Apel			

No	Nama citra	Citra Input	Citra non Bayangan grayscale	Citra <i>ground truth</i>
Citra bayangan dengan objek pembentuknya				
20.	Bangku batu			
21.	Kursi taman			
22.	Bunga			
23.	Payung			

No	Nama citra	Citra Input	Citra non Bayangan grayscale	Citra <i>ground truth</i>
Citra bayangan dengan objek pembentuknya				
24.	Tiang Biru			
Citra bayangan objek warna mirip bayangan				
25.	Batu1			
26.	Ember batu			
27.	Ember			

No	Nama citra	Citra Input	Citra non Bayangan <i>grayscale</i>	Citra <i>ground truth</i>
Citra bayangan objek warna mirip bayangan				
28.	Batu2			
29.	Batu3			
30.	Batu4			
Citra bayangan pada kerusakan jalan				
31.	Luban gl			

No	Nama citra	Citra Input	Citra non Bayangan grayscale	Citra <i>ground truth</i>
Citra bayangan pada kerusakan jalan				
32.	Luban g2			
33.	Luban g3			
34.	Luban g4			
35.	Luban g5			

No	Nama citra	Citra Input	Citra non Bayangan <i>grayscale</i>	Citra <i>ground truth</i>
Citra bayangan pada kerusakan jalan				
36.	Lubang6			

5.2. Hasil Uji Coba

Data-data pada tabel 5.1 diproses sesuai dengan penjelasan pada bab sebelumnya. Dari proses tersebut didapat hasil yang bervariasi. Sub-bab ini menjelaskan hasil yang didapat.

Citra-citra diproses dengan ε atau nilai epsilon yang berbeda yaitu 0,001 , 0,01 dan 0,1. Nilai epsilon tersebut didapatkan dari jurnal Graham D. Finlayson dengan judul *Intrinsic Images by Entropy Minimization* [20]. Masing-masing nilai ε dihitung RMSE untuk bagian bayangan , non bayangan dan total atau keseluruhan citra tanpa mempertimbangkan daerah bayangan atau non bayangan. Dari hasil *color invariant* , akan dicari nilai RMSE terkecil untuk mengetahui kinerja dari metode *color invariant* dalam mereduksi bayangan pada citra digital. Selain perbedaan nilai epsilon dalam pengujian ini juga menggunakan beberapa variasi kondisi citra seperti yang dijabarkan pada 3.2.1 sampai 3.2.4. Hal tersebut dilakukan untuk mengetahui pada kondisi citra mana metode *color invariant* bekerja paling optimal. Semakin kecil nilai RMSE, maka semakin bagus reduksi bayangan pada sebuah citra tersebut. Pertimbangan menghitung RMSE pada 3 bagian pada citra berdasar pada jurnal Saritha Murali et al dengan judul *A Survey on Shadow Removal Techniques for Single Image* [21].

5.2.1 *Background* citra bayangan cenderung abu-abu dan hitam

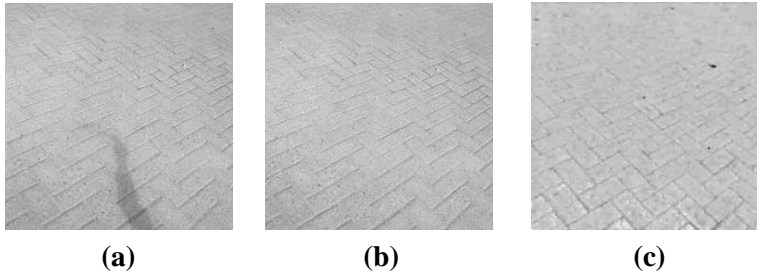
Hasil dari citra bayangan dengan *background* yang cenderung memiliki warna abu-abu dan hitam mendekati warna dari bayangan, dijabarkan dalam tabel 5.2 dibawah ini :

Tabel 5. 2 Hasil ketiga RMSE pada citra bayangan dengan *background* cenderung abu-abu dan hitam

No.	Nama Citra	ε	RMSE Bayangan	RMSE non Bayangan	RMSE Total	Waktu komputasi (second)
1.	Jalan	0,001	0,066	0,093	0,114	3,180
		0,01	0,066	0,093	0,114	3,153
		0,1	0,082	0,098	0,128	3,136
2.	Paving	0,001	0,013	0,064	0,066	3,204
		0,01	0,013	0,063	0,064	3,035
		0,1	0,021	0,101	0,103	3,053
3.	Keramik papan	0,001	0,040	0,141	0,147	2,993
		0,01	0,040	0,141	0,147	3,034
		0,1	0,040	0,141	0,147	2,960
4.	Keramik payung	0,001	0,053	0,148	0,157	2,812
		0,01	0,047	0,123	0,132	2,799
		0,1	0,089	0,190	0,210	2,814
5.	Dinding	0,001	0,213	0,230	0,313	2,905
		0,01	0,210	0,203	0,292	2,897
		0,1	0,156	0,121	0,197	2,935
6.	Paving rumput	0,001	0,105	0,201	0,226	2,903
		0,01	0,105	0,201	0,226	2,948
		0,1	0,091	0,150	0,176	2,914

Tabel 5.2 menunjukkan bahwa untuk citra bayangan dengan *background* cenderung abu-abu atau hitam, RMSE bayangan terkecil didapat pada citra nomor 2 dengan nilai ε atau epsilon 0,01. Begitu pula untuk RMSE non bayangan dan RMSE Total terkecil juga didapat pada citra nomor 2 dengan

nilai ε atau epsilon 0,01 . Sedangkan untuk waktu komputasi berkisar antara 2,8 sampai 3,1 *second*.



Gambar 5. 1 Perbandingan citra *input grayscale* (a) , citra tanpa bayangan *grayscale* (b) dan citra *output* (c) dengan RMSE bayangan, non bayangan dan total terkecil

5.2.2 *Background* citra bayangan berwarna

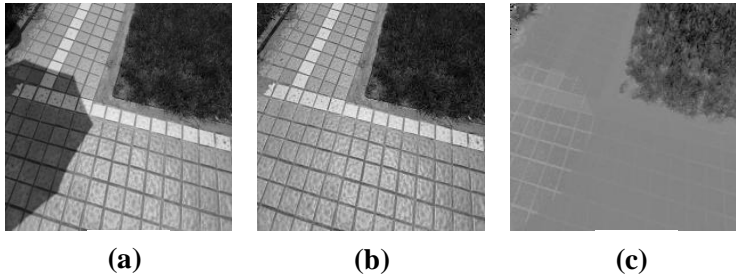
Hasil dari citra bayangan dengan *background* yang berwarna dan tidak menyerupai warna dari bayangan, dijabarkan dalam tabel 5.3 dibawah ini :

Tabel 5. 3 Hasil ketiga RMSE pada citra bayangan dengan *background* berwarna

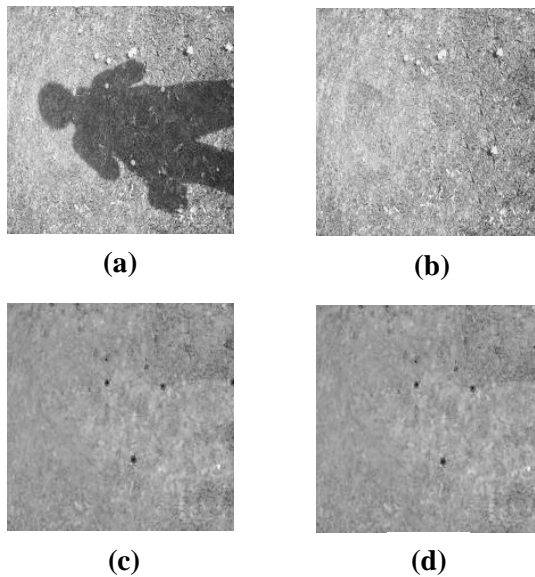
No .	Nama Citra	ε	RMSE Bayangan	RMSE non Bayangan	RMSE Total	Waktu komputasi (second)
1.	Batu rumput	0,001	0,165	0,212	0,269	3,077
		0,01	0,176	0,137	0,223	3,055
		0,1	0,176	0,137	0,223	3,246

No .	Nama Citra	ε	RMSE Bayangan	RMSE non Bayangan	RMSE Total	Waktu komputasi (second)
2.	Kayu rumput	0,001	0,113	0,187	0,218	2,955
		0,01	0,113	0,187	0,218	2,957
		0,1	0,113	0,187	0,218	3,056
3.	Rumput tanah	0,001	0,131	0,163	0,209	3,037
		0,01	0,121	0,144	0,188	3,124
		0,1	0,129	0,154	0,201	3,066
4.	Rumput	0,001	0,074	0,084	0,112	2,911
		0,01	0,074	0,084	0,112	3,112
		0,1	0,101	0,092	0,137	2,935
5.	Jalan rumput	0,001	0,197	0,249	0,318	2,941
		0,01	0,197	0,294	0,318	2,923
		0,1	0,095	0,115	0,149	3,777
6.	Keramik rumput	0,001	0,064	0,144	0,158	2,878
		0,01	0,073	0,140	0,158	2,894
		0,1	0,073	0,140	0,158	2,915

Tabel 5.3 menunjukkan bahwa untuk citra bayangan dengan *background* berwarna , RMSE bayangan terkecil didapat pada citra nomor 6 dengan nilai ε atau epsilon 0,001. Sedangkan untuk RMSE non bayangan dan RMSE total angka terkecil didapat pada citra nomor 4 dengan nilai ε atau epsilon 0,001 dan 0,01. Sedangkan untuk waktu komputasi berkisar antara 2,9 sampai 3,7 *second*.



Gambar 5. 2 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) dan citra *output* (c) dengan RMSE bayangan terkecil pada $\varepsilon=0,001$



Gambar 5. 3 Perbandingan citra *input* (a) , citra asli tanpa bayangan *grayscale* (b) , citra *output* pada $\varepsilon=0,001$ (c) dan $\varepsilon=0,01$ (d) dengan RMSE non bayangan dan RMSE Total terkecil

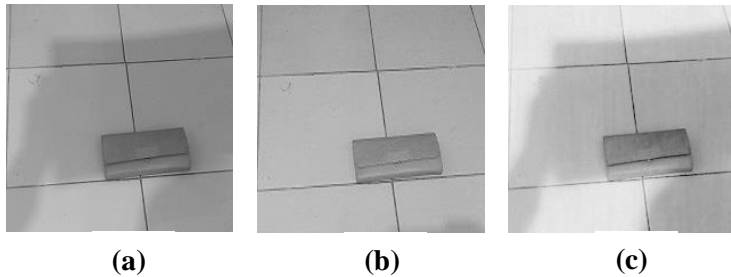
5.2.3 Citra bayangan dengan objek

Hasil dari citra bayangan dengan objek yang berada pada daerah bayangan, dijabarkan dalam tabel 5.4 dibawah ini :

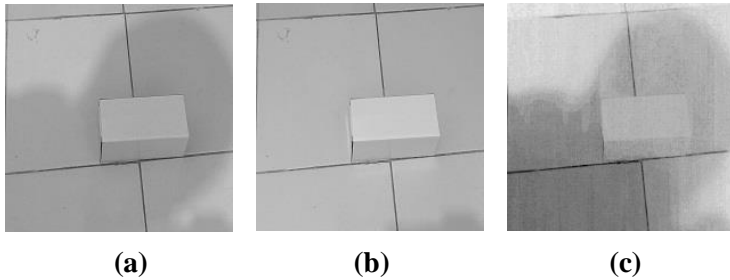
Tabel 5. 4 Hasil ketiga RMSE pada citra bayangan dengan objek

No .	Nama Citra	ε	RMSE Bayangan	RMSE non Bayangan	RMSE Total	Waktu komputasi (second)
1.	Kertas dompet	0,001	0,201	0,147	0,249	3,081
		0,01	0,196	0,151	0,248	3,122
		0,1	0,077	0,124	0,146	2,830
2.	Orang dompet	0,001	0,228	0,163	0,280	2,740
		0,01	0,226	0,161	0,278	2,713
		0,1	0,168	0,058	0,128	2,760
3.	Orang balok	0,001	0,200	0,157	0,254	2,820
		0,01	0,192	0,157	0,248	2,840
		0,1	0,142	0,079	0,163	2,872
4.	Kertas balok	0,001	0,178	0,130	0,220	2,911
		0,01	0,181	0,126	0,220	2,932
		0,1	0,181	0,126	0,280	2,912
5.	Kepala balok	0,001	0,165	0,182	0,245	2,800
		0,01	0,167	0,207	0,266	2,671
		0,1	0,145	0,075	0,163	3,011
6.	Orang bangku	0,001	0,222	0,285	0,361	2,934
		0,01	0,222	0,285	0,361	2,851
		0,1	0,222	0,285	0,361	2,910

Tabel 5.4 menunjukkan bahwa untuk citra bayangan dengan objek, RMSE bayangan terkecil didapat pada citra nomor 1 dengan nilai ε atau epsilon 0,001. Sedangkan untuk RMSE non bayangan dan RMSE total angka terkecil didapat pada citra nomor 2 dengan nilai ε atau epsilon 0,1. Pada kasus ini waktu komputasi berkisar antara 2,6 sampai 3,1 *second*. Terlihat pada gambar dibawah walaupun nilai RMSE kecil namun secara visual citra *output* bisa dikatakan buruk karena bayangan masih terlihat jelas. Untuk kasus lain nilai RMSE memang lebih besar namun hasil citra *output* sudah cukup bagus jika dinilai secara visual karena bayangan sudah tidak terlihat dengan jelas.

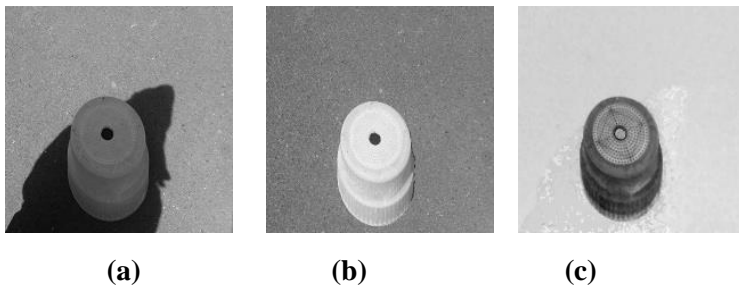


Gambar 5. 4 Perbandingan citra *input grayscale* (a), citra asli tanpa bayangan *grayscale* (b) dan citra *output* (c) dengan RMSE bayangan terkecil pada $\varepsilon=0,1$



Gambar 5. 5 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) dan citra *output* pada $\varepsilon=0,1$ (c).
RMSE non bayangan dan RMSE total terkecil

Berikut contoh kasus lain dimana bukan nilai RMSE terkecil namun secara visual bayangan sudah tidak terlihat dengan jelas. Contoh pada citra nomor 6 dengan nilai $\varepsilon = 0,001$.



Gambar 5. 6 Perbandingan citra asli dengan bayangan *grayscale* (a) citra asli tanpa bayangan (b) dan citra *output* pada $\varepsilon=0,001$ (c).

5.2.4 Citra bayangan dengan objek pembentuknya

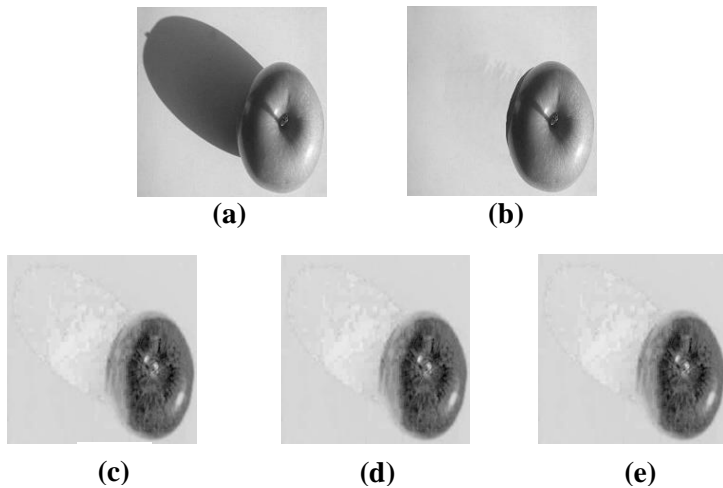
Hasil dari citra bayangan dengan objek pembentuk bayangan dilihat secara *visual* dan hanya ditampilkan nilai RMSE untuk daerah non bayangan saja. Hal tersebut dikarenakan citra *input* tanpa bayangan didapat dengan cara manual yaitu menggunakan *photoshop*. Nilai RMSE daerah non bayangan yang didapat dijabarkan dalam tabel 5.5 dibawah ini :

Tabel 5. 5 Hasil ketiga RMSE pada citra bayangan dengan objek pembentuknya

No	Nama Citra	ε	RMSE non Bayangan	Waktu komputasi (second)
1.	Apel	0,001	0,129	2,992
		0,01	0,129	2,712
		0,1	0,129	2,752
2.	Bangkubatu	0,001	0,205	3,091
		0,01	0,205	3,071
		0,1	0,186	3,041
3.	Kursitaman	0,001	0,166	3,091
		0,01	0,183	3,100
		0,1	0,183	3,471
4.	Bunga	0,001	0,278	2,851
		0,01	0,221	3,101
		0,1	0,221	3,661
5.	Payung	0,001	0,156	2,952
		0,01	0,222	3,091
		0,1	0,222	3,241

No	Nama Citra	ε	RMSE non Bayangan	Waktu komputasi (second)
6.	TiangBiru	0,001	0,189	3,031
		0,01	0,207	2,953
		0,1	0,207	3,300

Tabel 5.5 menunjukkan bahwa untuk citra bayangan dengan objek pembentuknya, RMSE bayangan, non bayangan dan Total terkecil didapat pada citra nomor 1 dengan nilai ε atau epsilon 0,001, 0,01 dan 0,1. Sedangkan untuk waktu komputasi berkisar antara 2,7 sampai 3,3 *second*.



Gambar 5. 7 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) dan citra *output* pada $\varepsilon=0,001$ (c) , $\varepsilon=0,01$ (d) dan $\varepsilon=0,1$ (e) . RMSE non bayangan terkecil

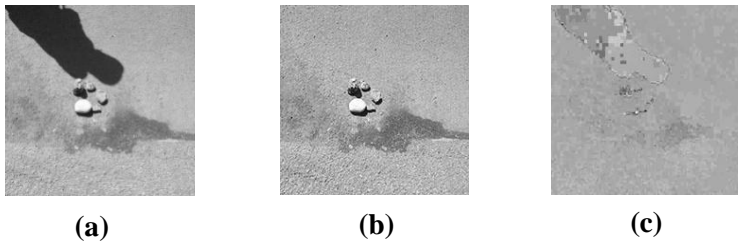
5.2.5 Citra bayangan dengan objek warna mirip bayangan

Hasil dari citra bayangan dengan objek warnanya mirip dengan warna bayangan, dijabarkan dalam tabel 5.6 dibawah ini :

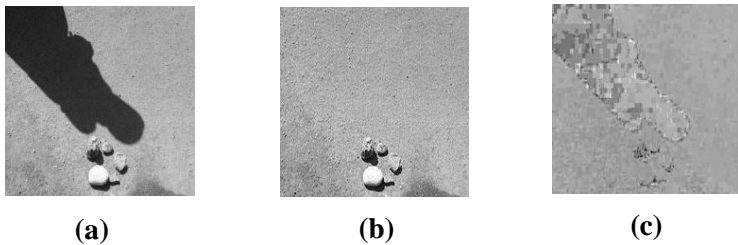
Tabel 5. 6 Hasil ketiga RMSE pada citra bayangan dengan objek warna mirip bayangan

No .	Nama Citra	ε	RMSE Bayangan	RMSE non Bayangan	RMSE Total	Waktu komputasi (second)
1.	Batu1	0,001	0,045	0,177	0,188	3,253
		0,01	0,128	0,196	0,234	2,770
		0,1	0,128	0,196	0,234	3,071
2.	Emberbatu	0,001	0,063	0,177	0,188	2,969
		0,01	0,062	0,174	0,185	2,974
		0,1	0,079	0,300	0,310	3,008
3.	Ember	0,001	0,080	0,170	0,188	3,312
		0,01	0,080	0,170	0,188	3,173
		0,1	0,081	0,225	0,240	3,076
4.	Batu2	0,001	0,031	0,077	0,083	2,808
		0,01	0,101	0,188	0,213	2,873
		0,1	0,098	0,153	0,182	3,243
5.	Batu3	0,001	0,123	0,137	0,184	2,860
		0,01	0,063	0,092	0,112	2,953
		0,1	0,063	0,092	0,112	2,810
6.	Batu4	0,001	0,070	0,075	0,103	2,844
		0,01	0,068	0,110	0,130	2,991
		0,1	0,175	0,195	0,262	3,037

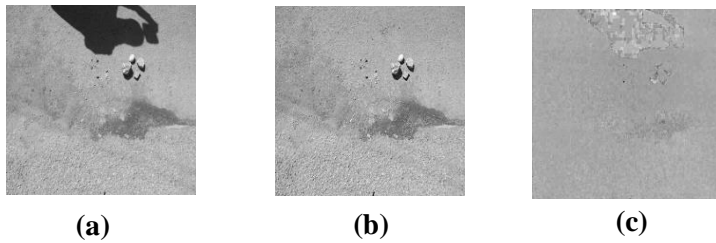
Tabel 5.6 menunjukkan bahwa untuk citra bayangan dengan objek warna mirip bayangan, RMSE bayangan terkecil didapat pada citra nomor 1 dengan nilai ε atau epsilon 0,001, RMSE non bayangan terkecil didapat pada citra nomor 6 dengan nilai ε atau epsilon 0,001 dan RMSE total terkecil didapat pada citra nomor 4 dengan nilai ε atau epsilon 0,001. Sedangkan untuk waktu komputasi berkisar antara 2,8 sampai 3,3 *second*.



Gambar 5. 8 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) dan citra *output* pada $\varepsilon=0,001$ (c) . RMSE bayangan terkecil



Gambar 5. 9 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) dan citra *output* pada $\varepsilon=0,001$ (c) . RMSE non bayangan terkecil



Gambar 5. 10 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) dan citra *output* pada $\varepsilon=0,001$ (c) . RMSE total terkecil

5.2.6 Citra bayangan pada kerusakan jalan

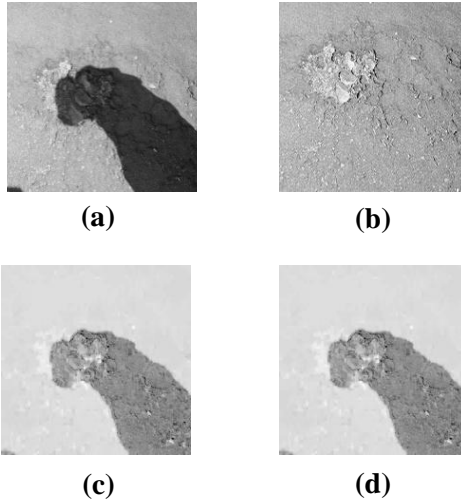
Hasil dari citra bayangan pada kerusakan jalan, dijabarkan dalam tabel 5.6 dibawah ini :

Tabel 5. 7 Hasil ketiga RMSE pada citra bayangan pada kerusakan jalan

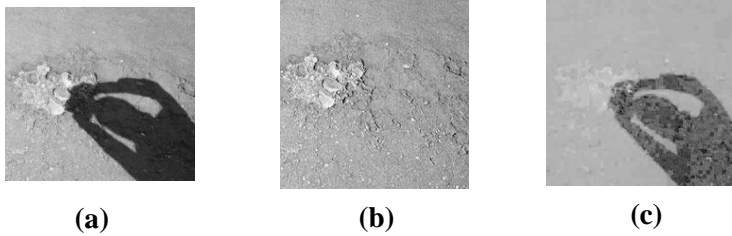
No .	Nama Citra	ε	RMSE Bayangan	RMSE non Bayangan	RMSE Total	Waktu komputasi (second)
1.	Lubang1	0,001	0,150	0,198	0,248	2,965
		0,01	0,150	0,198	0,248	3,072
		0,1	0,091	0,146	0,172	3,315
2.	Lubang2	0,001	0,111	0,173	0,206	2,883
		0,01	0,132	0,143	0,195	3,016
		0,1	0,132	0,143	0,195	3,230
3.	Lubang3	0,001	0,095	0,180	0,204	2,813
		0,01	0,081	0,176	0,194	3,041
		0,1	0,081	0,176	0,194	3,051
4.	Lubang4	0,001	0,119	0,117	0,167	2,815
		0,01	0,119	0,117	0,167	2,961
		0,1	0,140	0,079	0,161	3,033
5.	Lubang5	0,001	0,147	0,206	0,253	2,876
		0,01	0,147	0,206	0,253	3,069
		0,1	0,109	0,211	0,238	3,195
6.	Lubang6	0,001	0,123	0,233	0,263	2,791
		0,01	0,106	0,137	0,173	2,868
		0,1	0,106	0,137	0,173	2,937

Tabel 5.7 menunjukkan bahwa untuk citra bayangan pada kerusakan jalan, RMSE bayangan terkecil didapat pada

citra nomor 3 dengan nilai ε atau epsilon 0,01 dan 0,1 , RMSE non bayangan dan RMSE total terkecil didapat pada citra nomor 4 dengan nilai ε atau epsilon 0,1. Sedangkan untuk waktu komputasi berkisar antara 2,7 sampai 3,3 *second*.



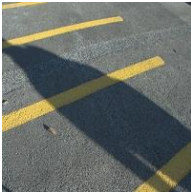




Gambar 5. 11 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) , citra *output* pada $\varepsilon=0,01$ (c) dan citra *output* pada $\varepsilon=0,01$ (d) . RMSE bayangan terkecil



Gambar 5. 12 Perbandingan citra *input grayscale* (a) , citra asli tanpa bayangan *grayscale* (b) dan citra *output* pada $\varepsilon=0,1$ (c) . RMSE non bayangan dan RMSE total terkecil

Berikut dilampirkan contoh perbandingan citra *input* , citra asli tanpa bayangan *grayscale* dan citra output dengan nilai ε atau epsilon berbeda. :

Tabel 5. 8 Perbandingan Citra *input*, Citra asli tanpa bayangan, dan Citra *output*

Citra <i>Input</i>	Citra asli tanpa bayangan <i>grayscale</i>	Nilai ε	Citra <i>output</i>
		0,001	
		0,01	
		0,1	

Untuk hasil citra *output* lengkap sebanyak 24 citra terlampir pada halaman lampiran.

BAB VI

KESIMPULAN DAN SARAN

Bab ini menjabarkan kesimpulan yang didapat dari penelitian penghapusan bayangan pada citra digital dengan menggunakan metode *Color Invariant* dan saran untuk penelitian di masa mendatang yang berhubungan.

6.1. Kesimpulan

Kesimpulan yang didapat setelah melakukan uji coba dan mendapatkan hasil dari penelitian penghapusan bayangan menggunakan metode *Color Invariant* adalah sebagai berikut :

1. Tugas akhir ini melakukan implementasi penghapusan bayangan pada citra digital dengan menggunakan metode *Color Invariant*. Langkah yang dilakukan adalah melakukan penormalan pada tiap *channel* R,G, dan B pada sebuah citra lalu dicari nilai log dari tiap hasil penormalan lalu dengan minimasi entropi citra bayangan akan diubah menjadi citra *illumination Invariant* 1D. Pada tahap ini sebenarnya bayangan sudah tereduksi atau terhapus namun agar tampilan citra menjadi lebih bagus citra tersebut akan diubah menjadi citra kromatisitas. Tetapi , citra kromatisitas tersebut tidak dapat diolah untuk perhitungan RMSE , sehingga citra kromatisitas tersebut diubah menjadi citra *grayscale*. RMSE dihitung berdasarkan citra asli tanpa bayangan *grayscale* dan citra *output*.
2. Pada tugas akhir tentang penghapusan bayangan ini, dibagi menjadi empat kasus yaitu citra bayangan dengan *background* hitam atau abu-abu, citra bayangan dengan *background* berwarna, citra bayangan dengan objek dan citra bayangan dengan objek pembentuknya. Keempat kasus tersebut diproses dengan nilai epsilon (ϵ) 0,001 ,

0,01 dan 0,1. Hasil terbaik diperoleh pada percobaan citra bayangan dengan *background* hitam atau abu-abu dengan nilai RMSE bayangan terkecil 0,013, RMSE non bayangan terkecil 0,063 dan RMSE total terkecil 0,064 dengan nilai epsilon (ϵ) 0,01. Nilai RMSE terbaik adalah yang paling mendekati 0 dari rentang 0 sampai 1. Untuk Waktu komputasi metode ini memerlukan waktu berkisar 2,7 sampai 3,7 *second*. Sehingga dari hasil penelitian ini dapat disimpulkan metode *Color Invariant* cukup optimal untuk penghapusan bayangan pada citra tunggal.

6.2. Saran

Adapun saran untuk penelitian selanjutnya yang berhubungan dengan penghapusan bayangan dengan menggunakan metode *Color Invariant* pada masa mendatang adalah sebagai berikut :

1. Melakukan peningkatan kontras pada citra *input* sebagai langkah *pre-processing* agar daerah bayangan lebih mudah dikenali.
2. Menggabungkan metode *Color Invariant* ini dengan metode lain agar bisa diperoleh *shadow mask* yang bisa menjadi jalan keluar agar citra *output* dapat dikembalikan ke dalam citra RGB. Selain itu juga dapat digabungkan dengan metode lain untuk pendeteksian bayangan agar hasil *output* lebih bagus.

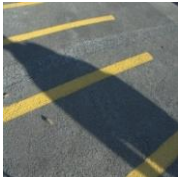



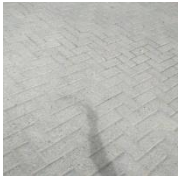
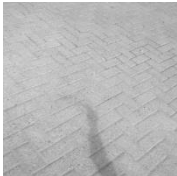

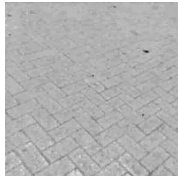
DAFTAR PUSTAKA

- [1] Qiang He, Chee-Hung Henry Chu , “*A New Shadow Removal Method for Color Images*”, *Advances in Remote Sensing*, USA, 2013
- [2] Zhang Yali, dkk., “ *Shadow Removal Of Single Texture Region Using Histogram Matching and Color Model Recovery*”, *The key laboratory of Integrated Microsystems*, Shenzhen Graduate School, Peking University, 2014
- [3] Abbas Nasrabadi, Mahdi Vaezi ,“ *Shadow Removal using Texture and Color Clustering*”, *International Journal of Environment and Science Education*, Iran, 2017
- [4] Cheng Lu , “*Removing Shadows from Color Image*”, Simon Fraser University, Kanada, 2006
- [5] Shraddha Singhal dan Sudha Radhika , “*Automatic Detection of Buildings from Aerial Images Using Color Invariant Features and Canny Edge Detection*” , India , 2014
- [6] D. Putra, “*Pengolahan Citra Digital*”, Yogyakarta: ANDI OFFSET, 2010.
- [7] Vijay Chondagar et al, “*A Review: Shadow Detection and Removal*”, (IJCSIT) *International Journal of Computer Science and Information Technologies*, Vol. 6 (6) , 5536-5541, 2015
- [8] Yongzhen Ke, Fan Qin, Weidong Min, dan Guiling Zhang, “*Exposing Image Forgery by Detecting Consistency of Shadow*”. *The Scientific World Journal*. pp 1-9, 2014.

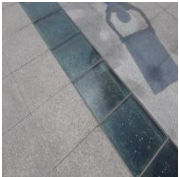


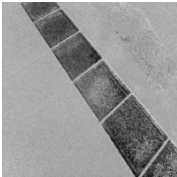
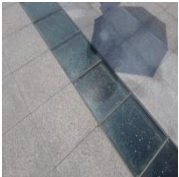


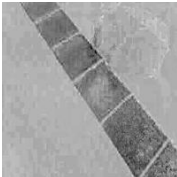
- [9] G.D. Finlayson, B. Schiele, dan J.L. Crowley, "Comprehensive Colour Image Normalization", *European Conference on Computer Vision 1998*, 1998.
- [10] G.D. Finlayson dan RuiXia Xu, "Non-Iterative Comprehensive Normalization", *first European Conference on Color in Graphics, Image and Vision*, halaman 134-139, 2002.
- [11] B.V. Funt dan G.D. Finlayson, "Color Constant Color Indexing", *IEEE PAMI*, 17:522-529, 1995.
- [12] T.Gevers dan A.W.M. Smeulders, "Color-based Object Recognition", *Patt.Rec.*, 32:453-464, 1999.
- [13] G.D. Finlayson, S.S. Chatterjee, dan B.V. Funt, "Color Angle Invariants for Object Recognition", *3rd Color Imaging Conference : Color, Science, Systems and Applications*, halaman 44-47, *Society for Imaging Science & Technology (IS&T)/Society for Information Display (SID) Joint Conference*, 1995.
- [14] G.Healey dan D. Slater, "Global Color Constancy : Recognition of Objects by Use of Illumination Invariant Properties of Color Distributions", *J.Opt.Soc.Am.A* , 11:3003-3010, 1994.
- [15] G.D. Finlayson dan S. Ssstrunk, "Optimization for Hue Constant RGB Sensors", *10th Color Imaging Conference*, halaman 343-348, 2002.
- [16] Wang, Chaoli dan Han-Wei Shen, "Information Theory in Scientific Visualization.", *Entropy* 13 halaman 254-273, 2011.
- [17] Pulung Nurtantio Andono dkk, *Pengolahan Citra Digital*, Yogyakarta: CV. Andi Offset, 2017.

- [18] A. Krishnan, P. Jayadevan and J. V. Panicker, "*Shadow Removal from Single Image using Color Invariant Method*", *International Conference on Communication and Signal Processing (ICCSP)*, Chennai, pp. 0583-0587, 2017
- [19] Finlayson, Graham D., Mark S. Drew and Cheng Lu, "*Entropy Minimization for Shadow Removal*", *International Journal of Computer Vision* 85 : 35-57, 2009
- [20] Finlayson. Graham, Drew. Mark , Lu.Cheng, "*Intrinsic Images by Entropy Minimization*", 3023. 10.1007/978-3-540-24672-5_46, 2004
- [21] Murali. Saritha, V K. Govindan, Kalady. Saidalavi, "*A Survey on Shadow Removal Techniques for Single Image*", *International Journal of Image, Graphics and Signal Processing*. 8(12) : 38-46, 2016


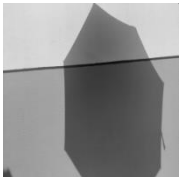
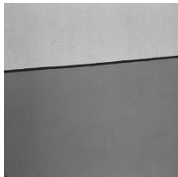
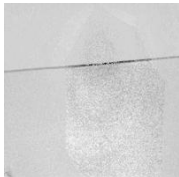
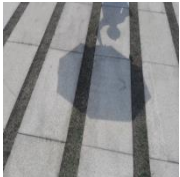


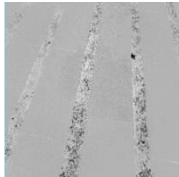
LAMPIRAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input cenderung abu-abu dan hitam</i>				
1.				
2.				


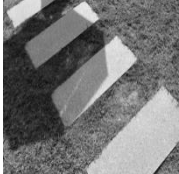
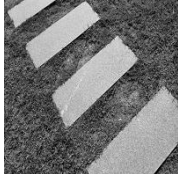
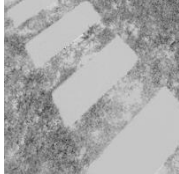



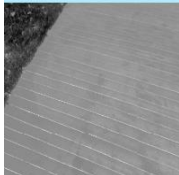
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	Citra input	Citra input grayscale	Citra grayscale non bayangan	Citra output
<i>Background citra input cenderung abu-abu dan hitam</i>				
3.				
4.				


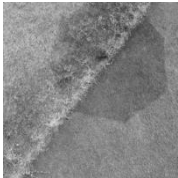
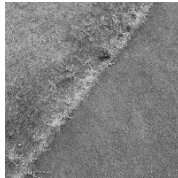
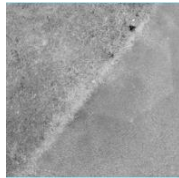


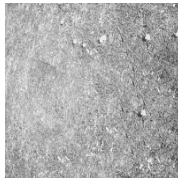
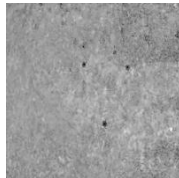
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	Citra input	Citra input grayscale	Citra grayscale non bayangan	Citra output
<i>Background citra input cenderung abu-abu dan hitam</i>				
5.				
6.				




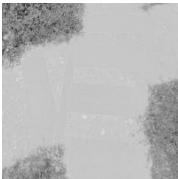

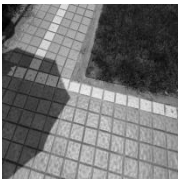

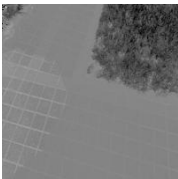
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale</i> non bayangan	Citra <i>output</i>
<i>Background citra input berwarna</i>				
7,				
8.				









LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input berwarna</i>				
9.				
10.				


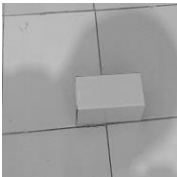






LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
<i>Background citra input berwarna</i>				
11.				
12.				

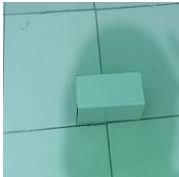
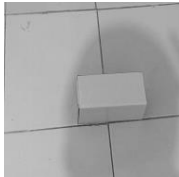






LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek				
13.				
14.				








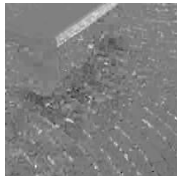
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra Output</i>
Citra bayangan dengan objek				
15.				
16.				






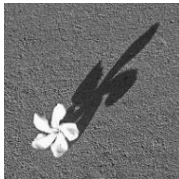
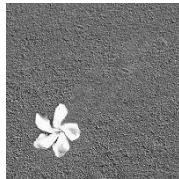
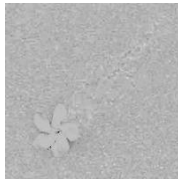
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek				
17.				
18.				

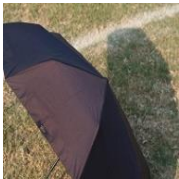







LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek pembentuknya				
19.				
20.				




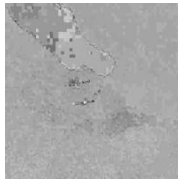




LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek pembentuknya				
21.				
22.				




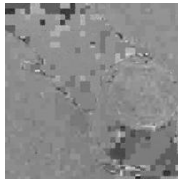
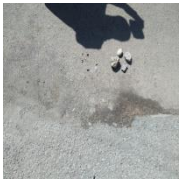


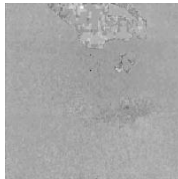
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek pembentuknya				
23.				
24.				

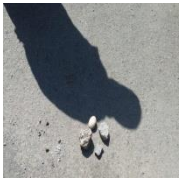



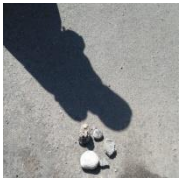



LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek warna mirip bayangan				
25.				
26.				



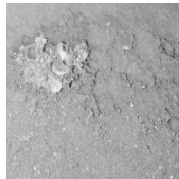
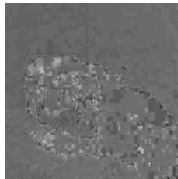



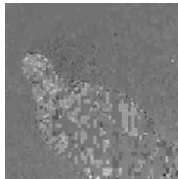
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek warna mirip bayangan				
27.				
28.				




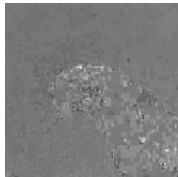




LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek warna mirip bayangan				
29.				
30.				




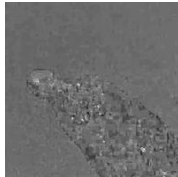



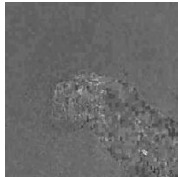
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan pada kerusakan jalan				
31.				
32.				

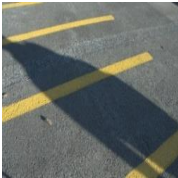



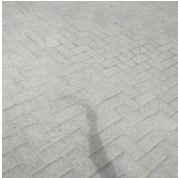
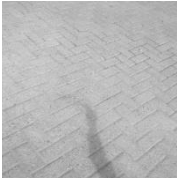


LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan pada kerusakan jalan				
33.				
34.				




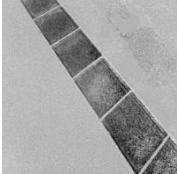



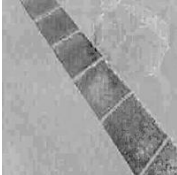
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,001$

No.	$\varepsilon = 0,001$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan pada kerusakan jalan				
35.				
36.				


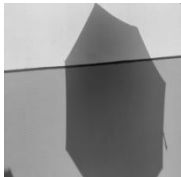
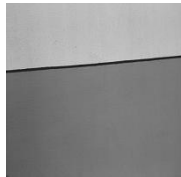
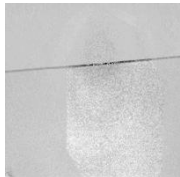
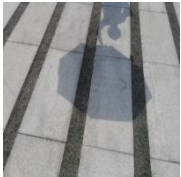


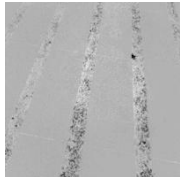
LAMPIRAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input cenderung abu-abu dan hitam</i>				
1.				
2.				


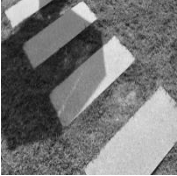
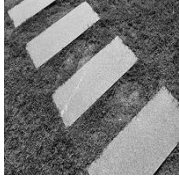
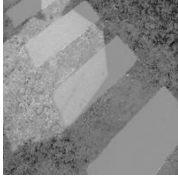




LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	Citra input	Citra input grayscale	Citra grayscale non bayangan	Citra output
<i>Background citra input cenderung abu-abu dan hitam</i>				
3.				
4.				


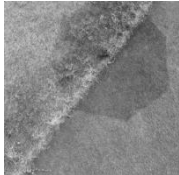
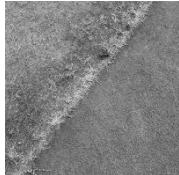
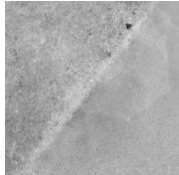


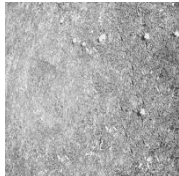
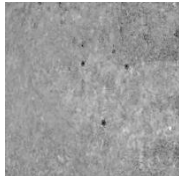
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	Citra input	Citra input grayscale	Citra grayscale non bayangan	Citra output
<i>Background citra input cenderung abu-abu dan hitam</i>				
5.				
6.				




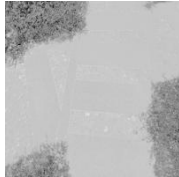

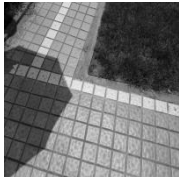


LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
<i>Background citra input berwarna</i>				
7.				
8.				









LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input berwarna</i>				
9.				
10.				


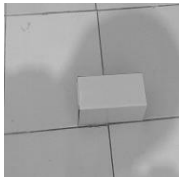






LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
<i>Background citra input berwarna</i>				
11.				
12.				

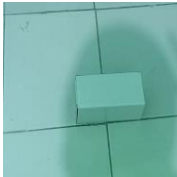
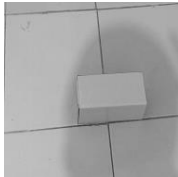






LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek				
13.				
14.				









LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek				
15.				
16.				




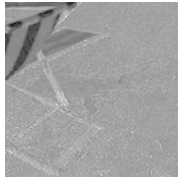


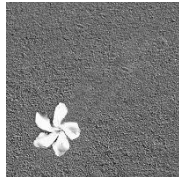

LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek				
17.				
18.				

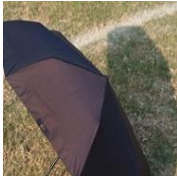







LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek pembentuknya				
19.				
20.				








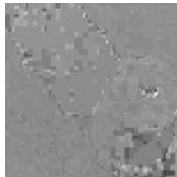
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek pembentuknya				
21.				
22.				




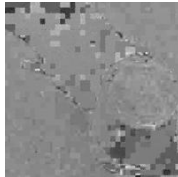
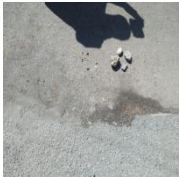



LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek pembentuknya				
23.				
24.				

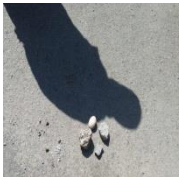


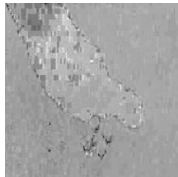
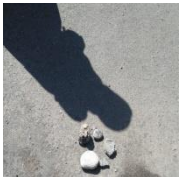


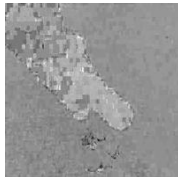
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek warna mirip bayangan				
25.				
26.				

LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$



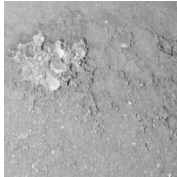
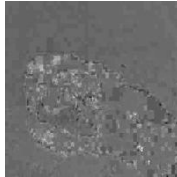



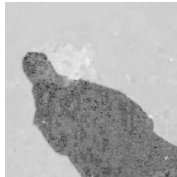
No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek warna mirip bayangan				
27.				
28.				

LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$



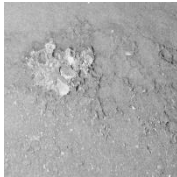
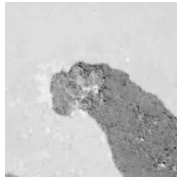




No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek warna mirip bayangan				
29.				
30.				

LANJUTAN




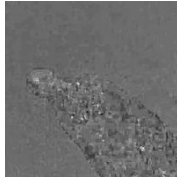



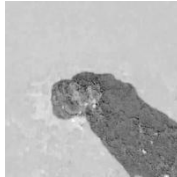
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan pada kerusakan jalan				
31.				
32.				

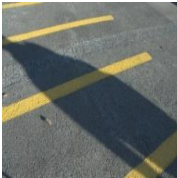



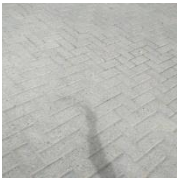
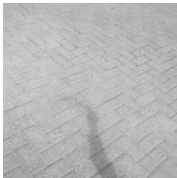


LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan pada kerusakan jalan				
33.				
34.				

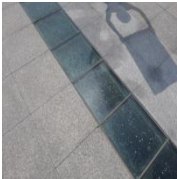


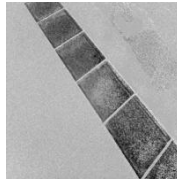

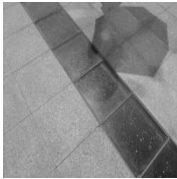
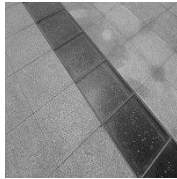

LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,01$

No.	$\varepsilon = 0,01$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan pada kerusakan jalan				
35.				
36.				


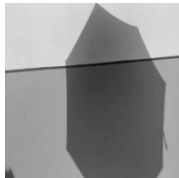
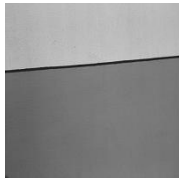
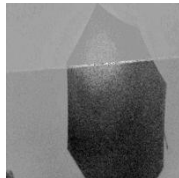
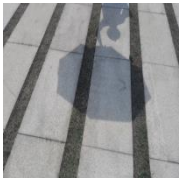



LAMPIRAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input cenderung abu-abu dan hitam</i>				
1.				
2.				


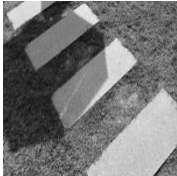
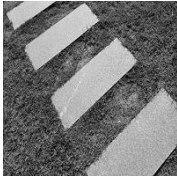
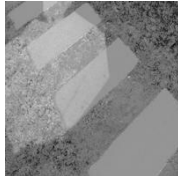

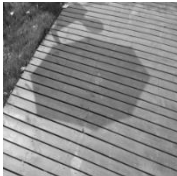

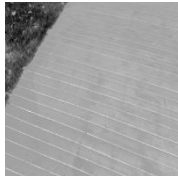
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input cenderung abu-abu dan hitam</i>				
3.				
4.				


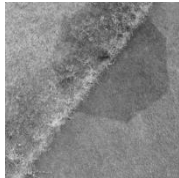
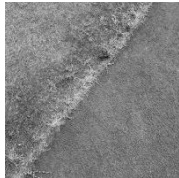
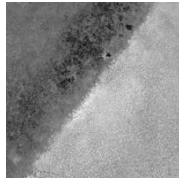


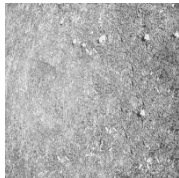
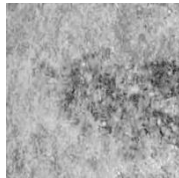
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input cenderung abu-abu dan hitam</i>				
5.				
6.				



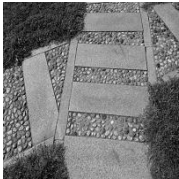
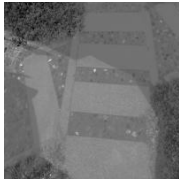

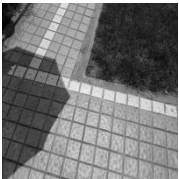


LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input berwarna</i>				
7.				
8.				









LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input berwarna</i>				
9.				
10.				


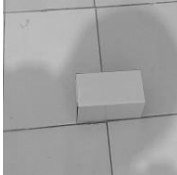

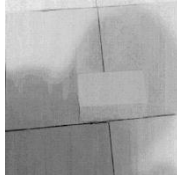




LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
<i>Background citra input berwarna</i>				
11.				
12.				

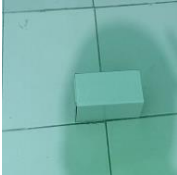
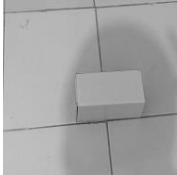

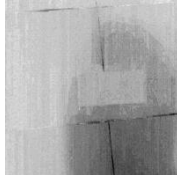

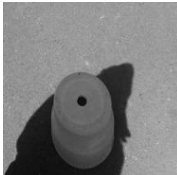


LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	<i>Citra input</i>	<i>Citra input grayscale</i>	<i>Citra grayscale non bayangan</i>	<i>Citra output</i>
Citra bayangan dengan objek				
13.				
14.				









LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek				
15.				
16.				






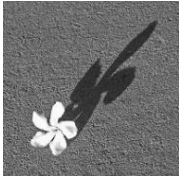
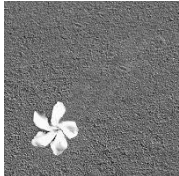

LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek				
17.				
18.				

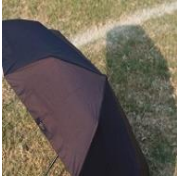
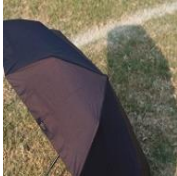






LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek pembentuknya				
19.				
20.				



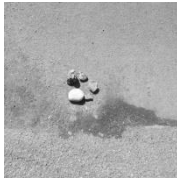




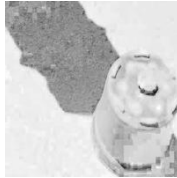
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek pembentuknya				
21.				
22.				




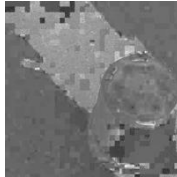




LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek pembentuknya				
23.				
24.				

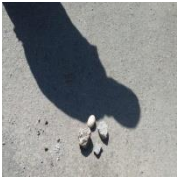


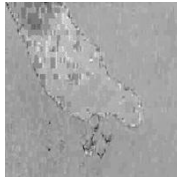
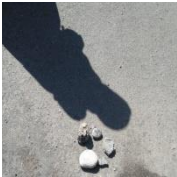



LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,001$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek warna mirip bayangan				
25.				
26.				



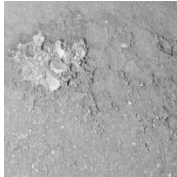
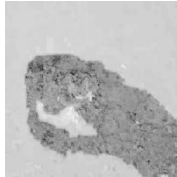



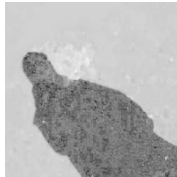
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangandegan objek warna mirip bayangan				
27.				
28.				




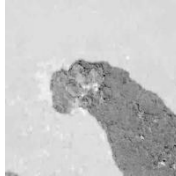




LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan dengan objek warna mirip bayangan				
29.				
30.				




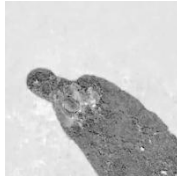



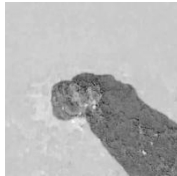
LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan pada kerusakan jalan				
31.				
32.				

LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan pada kerusakan jalan				
33.				
34.				

LANJUTAN
Hasil Uji Coba pada $\varepsilon = 0,1$

No.	$\varepsilon = 0,1$			
	Citra <i>input</i>	Citra <i>input grayscale</i>	Citra <i>grayscale non bayangan</i>	Citra <i>output</i>
Citra bayangan pada kerusakan jalan				
35.				
36.				

TENTANG PENULIS



Penulis memiliki nama panjang Saskia Rizkiana Putri. Lahir di Magetan tanggal 19 April 1998. Penulis pernah menempuh pendidikan di SD Muhammadiyah 1 Ngawi (2004-2010), SMP Negeri 2 Ngawi (2010-2013), dan SMA 2 Ngawi (2013-2016). Hobi penulis adalah membaca cerita pendek.

Penulis bercita-cita menjadi orang sukses dan dapat membuat orang tua bangga. Penulis aktif sebagai Staff *Entrepreneur Development* HIMATIKA ITS 2017-2018 dan 2018-2019. Penulis juga pernah melakukan magang atau kerja praktek di Perum LPPNPI Airnav Indonesia Cabang Surabaya dan ditempatkan pada bagian pengolahan data penerbangan. Selain itu, Penulis juga mengikuti program magang mahasiswa bersertifikat dari BUMN dan ditempatkan di BRI Kantor Wilayah Surabaya. Di Departemen Matematika ini, penulis mengambil rumpun Ilmu Komputer dan ingin menjadi Analis di suatu perusahaan *StartUp* ketika lulus nanti. Untuk informasi, kritik atau saran lebih lanjut dapat disampaikan melalui *e-mail* penulis di saskiaryzki99@gmail.com.