

# Pembangkitan Pola Batik dengan Menggunakan *Neural Transfer Style* dengan Penggunaan *Cost* Warna

<http://dx.doi.org/10.28932/jutisi.v6i2.2698>

Yosef Ariyanto Irawan<sup>#1</sup>, Andreas Widjaja<sup>✉\*2</sup>

<sup>#</sup> Program Studi Magister Ilmu Komputer, Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Surya Sumantri No.65, Bandung 40164, Indonesia

<sup>1</sup>1879005@maranatha.ac.id

<sup>2</sup>andreas.widjaja@it.maranatha.edu

**Abstract** — In this research, the neural transfer styles technique was applied to transfer styles of pattern of Batik, a traditional Indonesian cloth painted using the wax-resist dyeing technique, to some certain images. The transfer was performed using a well-known convolutional neural network (CNN) architecture. Some neural transfer tests were done to produce solid color which originally came from color clustered images. The color cost function of the CNN was computed at every epoch of the iterative neural computation. The result of the transfer are images with clustered colors and a slightly apparent color gradient. The produced images can be classified as "Creative Batik".

**Keywords**— Batik; Convolutional Neural Network; Color cluster; Neural transfer style.

## I. PENDAHULUAN

Batik merupakan salah satu budaya Indonesia. Batik adalah kain bergambar yang pembuatannya secara khusus dengan menuliskan atau menerakan malam pada kain itu, kemudian pengolahannya diproses dengan cara tertentu yang memiliki kekhasan. Batik Indonesia, sebagai keseluruhan teknik, teknologi, serta pengembangan motif dan budaya yang terkait, oleh UNESCO telah ditetapkan sebagai Warisan Kemanusiaan untuk Budaya Lisan dan Non-bendawi (*Masterpieces of the Oral and Intangible Heritage of Humanity*) sejak 2 Oktober 2009 [1]. Batik di Indonesia terdapat lebih dari 22 jenis.

Batik biasanya dibuat oleh seniman - seniman batik. Tidak semua orang dapat menggambar batik. Untuk melukis batik dibutuhkan kemampuan khusus. Setiap seniman yang dapat menggambar batik belum tentu dapat menggambar beberapa jenis batik. Setiap batik memiliki ciri khas sendiri - sendiri. Selain itu untuk membuat sebuah batik, seniman akan memerlukan waktu yang berbulan - bulan untuk

membuat batik. Jika seniman hanya untuk mencoba-coba, maka hal tersebut akan memakan waktu yang banyak.

Jika gambar kita dapat diubah dengan memasukan pola batik ke dalam gambar, akan mempermudah desainer ingin merancang batik. Desainer tidak perlu merancang sketsa batik terlebih dahulu jika ingin menggunakan sebuah *style* pada sebuah gambar. Desainer juga dapat bereksperimen dengan gambar-gambar dengan menggunakan pola batik dengan sangat mudah. Hal tersebut akan sangat membantu.

Pada tahun 2015, Gatys, Ecker, dan Matthias Bethge [2] mengeluarkan paper tentang *Neural Transfer Style*. *Neural Transfer Style* adalah pengimplementasian dari CNN untuk penggunaan memindahkan *style* dari gambar ke gambar konten. Setelah Gatys banyak orang yang mendalami algoritma ini. Gatys merancang algoritma tersebut dengan menggunakan lukisan untuk mengetesnya.

Batik memiliki ciri khas tersendiri dari pada lukisan. Pada umumnya batik memiliki warna yang *solid* (tidak ada gradasi warna). Masalah ini belum dirancang pada algoritma *Neural Transfer Style*. Jika melakukan perubahan pada arsitektur CNN tersebut, akan diperlukan pengujian untuk kinerja pada arsitektur CNN tersebut. Selain itu hasil batik dari algoritma tersebut dibutuhkan penilaian dari pakar untuk memastikan gambar hasil tersebut dapat disebut sebagai batik atau tidak. Mungkin arsitektur pada pengujian ini dapat bekerja dengan *style* dengan warna *solid* yang lain. Tetapi pada penelitian ini akan dilakukan mendalam pada batik.

Mengetahui arsitektur CNN yang dapat memindahkan pola batik dengan menampilkan warna yang *solid* (tidak ada gradasi warna), merupakan tujuan utama penelitian ini. Arsitektur tersebut harus diketahui kinerjanya saat memindahkan pola batik. Selain kinerja, penelitian ini juga akan mengetahui gambar yang dihasilkan dapat digolongkan menjadi batik atau tidak. Penilaian tersebut akan dinilai oleh pakar yang ahli pada batik.

Pada penelitian ini akan menggunakan hanya 5 gambar pola batik sebagai data pola. Sedangkan data konten akan menggunakan hanya 2 gambar yaitu gambar foto wajah dan foto pemandangan. Algoritma yang dibandingkan hanya dari algoritma dari Gatys dan algoritma yang ditambahi dengan *cost* warna.

## II. LANDASAN TEORI

### A. Pola batik

Pola batik adalah corak atau pola yang menjadi kerangka gambar pada batik berupa perpaduan antara garis, bentuk dan isen menjadi satu kesatuan yang mewujudkan batik secara keseluruhan. Motif-motif batik itu antara lain adalah motif hewan, manusia, geometris, dan motif lain. Motif batik sering juga dipakai untuk menunjukkan status seseorang. Membatik merupakan tradisi turun-menurun. Karena itu, sering motif batik menjadi ciri khas dari batik yang diproduksi keluarga tertentu [3].

### B. Neural Network

*Neural Network* adalah model yang terinspirasi oleh bagaimana neuron dalam otak manusia bekerja. Tiap neuron pada otak manusia saling berhubungan dan informasi mengalir dari setiap neuron tersebut. Tiap neuron menerima input dan melakukan operasi dot dengan sebuah *weight*, menjumlahkannya (*weighted sum*) dan menambahkan *bias*. Hasil dari operasi ini akan dijadikan parameter dari *activation function* yang akan dijadikan *output* dari neuron tersebut [4].

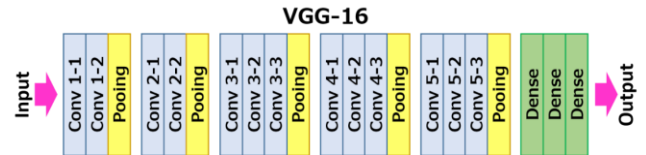
### C. Convolutional Neural Network

*Convolutional Neural Network* (CNN) adalah salah satu jenis *Neural Network* yang biasa digunakan pada data *image*. CNN bisa digunakan untuk mendeteksi dan mengenali objek pada sebuah *image* [5]. CNN memiliki kelebihan dibandingkan *Neural Network*. *Neural Network* kurang efisien untuk memproses gambar karena jumlah parameter yang terlalu banyak jika 1 piksel dijadikan 3 parameter karena RGB. Arsitektur dari CNN dibagi menjadi 2 bagian besar, *Feature Layer* dan *Fully-Connected Layer* (MLP) pada *Feature layer* biasanya terdapat *convolution layer* dan *pooling layer*. Pada *convolution layer* terdapat neuron-neuron yang melakukan filter pada gambar. Filter seperti untuk mendeteksi sudut atas atau mendeteksi yang lain pada gambar. *Pooling layer* biasanya berada setelah *conv. layer*. Pada prinsipnya *pooling layer* terdiri dari sebuah filter dengan ukuran dan *stride* tertentu yang akan bergeser pada seluruh area *feature map* [6].

*Pooling* yang biasa digunakan adalah *Max Pooling* dan *Average Pooling*. Sebagai contoh jika kita menggunakan *Max Pooling 2x2* dengan *stride 2*, maka pada setiap pergeseran filter, nilai maximum pada area *2x2* pixel tersebut yang akan dipilih, sedangkan *Average Pooling* akan memilih nilai rata-ratanya [7].

### D. VGG Network

*VGG network* adalah sebuah model CNN yang dibangun oleh Simonyan dan Zisserman [8] dari University of Oxford. Model tersebut memperoleh akurasi 92,7% pada *dataset ImageNet*. *Dataset* tersebut memiliki 14 juta gambar yang terbagi menjadi 1000 kelas. *VGG network* memiliki 13 *convolutional layer* dan 3 *classification layer* dengan 138 juta *parameter* [9].



Gambar 1 Arsitektur VGG network [9]

Pada Gambar 1 merupakan arsitektur *VGG network*, *VGG network* memiliki 13 *convolutional layer*, 5 *max pooling layer* dan 3 *classification layer*. *VGG network* membagi *convolutional layer* dengan *max pooling layer* sebagai pembatas menjadi 5 bagian. Bagian-bagian tersebut disebut "conv1", "conv2", "conv3", "conv4" dan "conv5". "conv1" dan "conv2" terdiri dari 2 *convolutional layer*. Sedangkan "conv3", "conv 4" dan "conv 5" terdiri dari 3 *convolutional layer*. Pada "conv1" setiap layer memiliki 64 filter. Pada "conv2" setiap layer memiliki 128 filter. Pada "conv3" setiap layer memiliki 256 filter. Pada "conv4" dan "conv 5" setiap layer memiliki 512 layer. Hal tersebut yang membuat *VGG network* memiliki 138 juta parameter.

### E. Neural Transfer Style

*Neural Transfer Style* Berbeda dengan GAN (*Generative Adversarial Networks*). GAN akan membangkitkan konten dari data training secara random. Sedangkan *Neural Transfer Style* memindahkan *style* ke konten gambar. Maka hasil *Neural Transfer Style* akan memiliki konten yang diinginkan. Sedangkan GAN memiliki konten gambar yang *random*.

*Neural Transfer Style* akan memindahkan *style* pada gambar *style* ke gambar konten. *Neural Transfer Style* akan memindahkan warna dan patern pada gambar *style* ke gambar konten. *Neural Transfer Style* akan memindahkan aksent-aksent detail pada sebuah *style image* tetapi akan mempertahankan konten pada gambar konten. Hanya saja warna yang dihasilkan tidak *solid* (tidak memiliki gradasi) seperti gambar *style*. Hasil pembangkitan pola juga menghasilkan pola gradasi.

Algoritma ini menggunakan *VGG network* yang sudah dilatih untuk *object recognition and localisation*, pada algoritma mereka memiliki 3 bagian yaitu, *content representation*, *style representation*, dan *style transfer*. Pada bagian *content representation* adalah untuk menghitung apakah konten pada input gambar dan gambar yang sudah dibuat memiliki nilai konten yang sama atau tidak [2].

CNN dapat memindahkan *style* dari gambar 1 ke konten gambar yang lain. Pada Algoritma ini memiliki perhitungan *Cost* untuk *style* dan perhitungan *cost* untuk konten yang terpisah. Maka kita dapat mengubah komposisi bobot antara *style cost* dan konten *cost* untuk mendapatkan gambar perceptual yang baru. Semakin kuat *style* yang dipindahkan pada gambar maka konten pada gambar tersebut akan hilang, begitu juga sebaliknya.

#### F. K-means

*K-means* adalah algoritma untuk membuat kluster dari suatu kumpulan data. Algoritma ini termasuk ke dalam *unsupervised learning*. Algoritma ini mencari titik tengah dari  $k$  kluster, dimana  $k$  adalah banyak kluster yang akan dicari. Algoritma ini menghitung jarak pada titik tengah kluster dengan data-data sekitar [10]. Untuk mendapatkan jarak-jarak yang paling dekat dengan jumlah kluster yang tidak terlalu banyak maka pada penelitian ini akan menggunakan *elbow method*. *Elbow method* adalah sebuah teknik untuk mencari jumlah kluster dengan cara mencoba 1 demi 1 jumlah kluster dan membandingkan SSE (*sum of squared error*) dari titik *center* dengan data-data yang terdekat [11]. Algoritma *K-means* ini akan digunakan untuk mencari warna yang dominan pada pola batik.

### III. TINJAUAN PUSTAKA

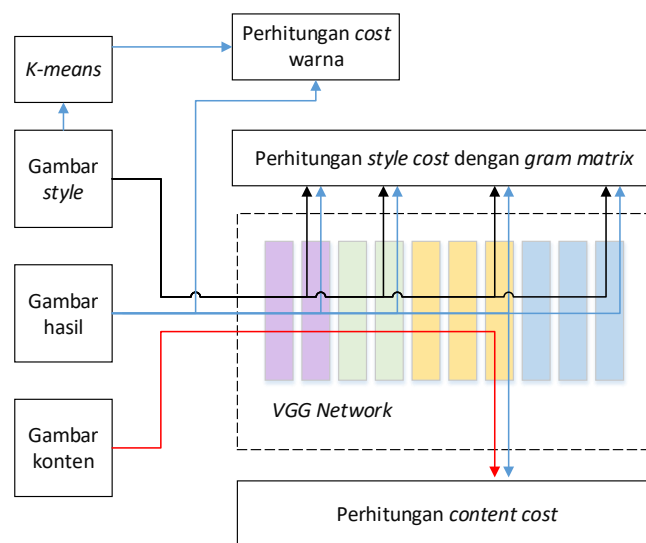
*Generate batik* akan menggunakan algoritma *Neural Transfer Style*. Pada *Neural Transfer Style* akan menggunakan salah satu penggunaan CNN. CNN memiliki banyak penggunaan. Menurut Schmidhuber [12], CNN banyak digunakan pada kasus visual deep learners. Awal mula digunakan pada MNIST untuk mendeteksi tulisan angka dengan tangan.

Sedangkan *Neural Transfer Style*, memiliki terobosan yang sangat bagus. Pada tahun 2003 N. Ashikhmin hanya focus pada memindahkan texture. Pada penelitian tersebut tidak memehatkan *edge orientation* dan warna *style*. Menurut Gatys, Ecker, dan Bethge [2] *Neural Transfer Style* dapat menciptakan gambar yang penuh arti dari menyatukan gambar konten dan gambar *style*. Gambar yang memiliki penuh arti dari pada penelitian 2003 yang hanya memindahkan texture saja

Menurut Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song [13] *Neural Transfer Style* sudah banyak yang melakukan penelitian tentang *Neural Transfer Style*. Dari penelitian-penelitian tersebut belum ada yang mencoba membuat warna lebih *solid* dan melakukannya pada batik.

Pada Algoritma *Neural Transfer Style* yang dikembangkan oleh Justin Johnson, [14] Justin Johnson menggunakan *Neural Network* yang berbeda dengan Gatys, pada algoritma *Fast Style Transfer* tidak menggunakan pengulangan yang digunakan pada *Neural Transfer Style*. Pada *Fast Style Transfer* menggunakan *Feed Forward*

*Neural Network*. Tetapi di penelitian ini akan menggunakan *Neural Transfer Style* yang menggunakan pengulangan karena untuk *training Feed Forward Neural Network* membutuhkan waktu lebih lama pada awal.



Gambar 2 Arsitektur CNN yang akan digunakan

Untuk mendapatkan warna yang lebih *solid* maka akan menggunakan arsitektur seperti pada Gambar 2. Pertama pola batik akan dimasukan pada algoritma *K-means* untuk mendapatkan kluster-kluster warna pada batik tersebut. Kluster-kluster warna tersebut akan digunakan pada perhitungan *cost* warna dengan gambar yang dihasilkan dari *Neural Transfer Style*. Perhitungan *cost* tersebut akan dilakukan setiap *epoch* pada proses learning.

*Cost* warna akan menghitung jarak antara warna yang dihasilkan terhadap kluster-kluster warna pada gambar *style*. Maka semakin jauh jarak warna yang dihasilkan terhadap kluster-kluster warna maka *total cost* akan menjadi lebih besar. Dengan Arsitektur CNN ini akan membuat warna-warna yang dihasilkan lebih dengan terhadap kluster-kluster warna pada gambar *style*.

$$Cost\ warna(k, x) = \frac{1}{n} \sum_{i=1}^n (K^2 - H^2) \quad (1)$$

Persamaan (1) merupakan rumus untuk menghitung *cost* untuk warna di mana  $K$  tersebut titik center kluster terdekat dan  $H$  merupakan 1 piksel yang dihasilkan dari *Neural Transfer Style*. Sedangkan  $n$  merupakan banyak nya piksel. Jadi inti dari rumus tersebut adalah mencari rata - rata jarak antara setiap piksel yang dihasilkan dari *Neural Transfer Style* dengan kluster warna terdekat dengan piksel tersebut. Sedangkan perhitungan *cost style* dan perhitungan *cost* untuk konten sama seperti pada *Neural Transfer Style*



$$\begin{aligned}
 \text{Cost total}(\vec{p}, \vec{a}, \vec{x}, \vec{k}) & \\
 &= \alpha \text{Cost konten}(\vec{p}, \vec{x}) \\
 &+ \beta \text{Cost style}(\vec{a}, \vec{x}) \\
 &+ \gamma \text{Cost warna}(\vec{k}, \vec{x})
 \end{aligned} \quad (2)$$

Persamaan (2) merupakan total *cost* yang telah ditambahkan *cost* warna didalamnya, dimana  $\gamma$  adalah weight faktor untuk *cost* warna untuk  $k$  merupakan kluster dan  $x$  merupakan gambar yang dihasilkan *Neural Transfer Style*. Jika *weight factor* ditingkatkan maka *cost* warna akan lebih berpengaruh pada proses *training*

#### IV. METODE PENELITIAN

Pada eksperimen yang dilakukan, digunakan 2 metode evaluasi. Dua metode itu adalah evaluasi kuantitatif dan evaluasi kualitatif. Kuantitatif akan berfokus pada *cost style*, *cost* konten dan *cost* warna. sedangkan kualitatif akan bergantung pada estetika penilai pengamat. Hasil evaluasi akan terkait ke banyak faktor (misal usia dan pekerjaan peserta). Eksperimen yang membandingkan perbedaan model *Neural Transfer Style* dilakukan secara kualitatif maupun kuantitatif.

##### A. Pengaturan Eksperimen

Eksperimen menggunakan 5 *style* batik dan 2 konten image. Untuk *style image* digunakan 5 gambar seperti pada Gambar 3. Batik - batik akan mewakili: karakter yang dibuat batik seperti nomor 1, pola batik yang berulang-ulang seperti nomor 2, karakter yang dibuat batik dan memiliki latar belakang seperti nomor 3, pola batik yang berulang dan memiliki karakter seperti nomor 4 dan pola batik yang menggambarkan pemandangan seperti nomor 5. Batik – batik ini didapatkan dari *web site* freepik.com. Gambar – gambar di *website* bebas untuk digunakan dan bersifat tidak komersial. Batik tersebut terbuat dari *vector*.



Gambar 3 Batik yang akan digunakan pada evaluasi

Untuk gambar konten akan menggunakan 2 gambar dari *dataset* “NPRgeneral” yang terdiri dari gambar-gambar yang mencakup berbagai karakteristik (misal: kontras, tekstur, tepi dan struktur yang bermakna) dan memiliki banyak kriteria. *Dataset* tersebut didapatkan dari *CvOnline website* [15]. Dua gambar tersebut seperti pada Gambar 4 dan Gambar 5. *Dataset* tersebut telah dievaluasi oleh Mould and Rosin. *Style Image* dan konten *Image* akan menggunakan ukuran gambar sebesar 512x512 piksel.



Gambar 4 Gambar konten 1 [15]



Gambar 5 Gambar konten 2 [15]

Setiap model dan algoritma *Neural Transfer Style* yang akan diuji coba akan dijalankan pada GPU yang sama. GPU yang akan digunakan adalah NVIDIA GTX 1060 dengan RAM 6GB DDR5 pada GPU. Setiap model dan algoritma akan diatur sebaik mungkin pada hyperparameter untuk memaksimalkan nilai *cost*. Setiap model dan algoritma akan dibuat dengan menggunakan PyTorch.

##### B. Kuantitas Evaluasi

Kuantitas evaluasi akan dilakukan dengan membandingkan arsitektur yang menggunakan perhitungan *cost* warna dengan arsitektur yang tidak menggunakan perhitungan *cost* warna. Perbandingan tersebut akan melihat dari sisi kecepatan penurunan *cost style* dan konten, *cost* warna, dan *histogram* warna gambar hasil. Dengan menggunakan beberapa parameter tersebut diharapkan dapat mengetahui seberapa besar pengaruh dan dampak dengan menggunakan perhitungan *cost* warna.

Pada evaluasi untuk mengetahui perbandingan – perbandingan tersebut maka dilakukan dengan beberapa pengujian. Pengujian-pengujian tersebut dilakukan dengan kondisi yang berbeda- beda. Kondisi tersebut akan dijelaskan pada sub bab berikut.

1) *Pengujian penggunaan cost warna.* Pada pengujian ini akan membandingkan perhitungan dengan menggunakan *cost* warna dengan yang tidak menggunakan *cost* warna. Pengujian ini bertujuan untuk melihat pengaruh dan dampak memasukan *cost* warna pada fungsi *cost*. Pengujian ini akan diamati dari sisi kecepatan penurunan *cost style* dan konten, penurunan *cost* warna dan *histogram* warna yang dihasilkan

TABEL I  
HYPERPARAMETER PADA PENGUJIAN PENGGUNAKAN COST WARNA

Nama Hyperparameter	Nilai 1	Nilai 2
Jumlah kluster	2	2
Weight factor untuk cost warna	50	-
Weight factor untuk cost konten	10	10
Weight factor untuk style	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>
Jumlah iteration	10000	10000

Pada pengujian ini menggunakan pola batik nomor 1 pada Gambar 2 dan konten pada Gambar 3. Pengujian ini menggunakan pengaturan *hyperparameter* seperti pada Tabel I. Pada tabel I terdapat kolom Nilai 1 dan Nilai 2, kedua nilai itu yang akan di bandingkan pada pengujian ini. Nilai 1 yang menggunakan perhitungan *cost* warna sedangkan nilai 2 tidak menggunakannya. Untuk *weight factor* konten dan *style*, pengujian ini menggunakan *default value* dari algoritma tersebut. Nilai tersebut paling sesuai untuk rata-rata gambar yang digunakan. Untuk *weight factor cost* warna, pada pengujian ini digunakan nilai 50 dan akan coba ditingkatkan pada pengujian berikutnya. Untuk jumlah *iteration* akan menggunakan 10000 karena pada *epoch* 10000 algoritma sudah mendapatkan diagram *total cost* yang rata atau penurunannya hanya sedikit. Pengujian menggunakan jumlah kluster 2. Jumlah kluster tersebut didapat dengan *Elbow Method* dari pola batik nomor 1 pada Gambar 2

2) *Pengujian pengaruh peningkatan jumlah kluster warna.* Pada pengujian pengaruh peningkatan jumlah kluster warna akan dilakukan dengan membandingkan pengujian dengan jumlah kluster yang berbeda. Pengujian ini akan dilakukan dengan hanya perbedaan terletak pada jumlah kluster warna sedangkan *hyperparameter* yang lain sama. Maka *hyperparameter* akan seperti pada Tabel II. Pada pengujian ini menggunakan pola batik nomor 1 pada Gambar 2 dan konten pada Gambar 3. Sampel Nilai 1 masih menggunakan jumlah kluster yang sama dari *elbow method* yang sama. Pengujian ini ingin melihat dampak dengan meningkatkan jumlah kluster warna.

TABEL II  
HYPERPARAMETER PADA PENGUJIAN PENGARUH PENINGKATAN JUMLAH KLUSTER WARNA

Nama Hyperparameter	Nilai 1	Nilai 2
Jumlah kluster	3	2
Weight factor untuk cost warna	50	50
Weight factor untuk cost konten	10	10
Weight factor untuk style	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>
Jumlah iteration	10000	10000

3) *Pengujian pengaruh peningkatan weight factor warna.* Pada pengujian pengaruh peningkatan *weight factor* warna dilakukan dengan membandingkan pengujian dengan *weight factor* warna yang berbeda. Pengujian ini dilakukan dengan hanya perbedaan terletak pada *weight factor* warna sedangkan *hyperparameter* yang lain sama. Maka *hyperparameter* akan seperti pada Tabel III. Pengujian ini ingin melihat dampak dengan meningkatkan *weight factor* warna Jumlah kluster masih menggunakan nilai yang sama dari *elbow method* yang sama karena masih menggunakan pola batik no 1.

TABEL III  
HYPERPARAMETER PADA PENGUJIAN PENGARUH PENINGKATAN WEIGHT FACTOR WARNA

Nama Hyperparameter	Nilai 1	Nilai 2	Nilai 3
Jumlah kluster	3	3	3
Weight factor untuk cost warna	50	100	1000
Weight factor untuk cost konten	10	10	10
Weight factor untuk style	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>
Jumlah iteration	10000	10000	10000

4) *Pengujian dengan pola batik yang lain.* Pada pengujian dengan pola batik yang lain dilakukan dengan membandingkan hasil dari arsitektur yang menggunakan perhitungan *cost* warna dengan arsitektur yang tidak menggunakan perhitungan *cost* warna pada setiap pola batik. Tetapi karena setiap pola memiliki komposisi warna yang berbeda-beda maka jumlah kluster akan berbeda-beda juga. Setiap pola batik akan ditentukan jumlah klusternya dengan menggunakan *elbow method*.

TABEL IV  
HYPERPARAMETER PADA PENGUJIAN PENGGUNAAN COST WARNA UNTUK BATIK YANG LAIN

Nama Hyperparameter	Nilai 1	Nilai 2
Weight factor untuk cost warna	50	-
Weight factor untuk cost konten	10	10
Weight factor untuk style	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup> , 1000/ 512 <sup>2</sup>	1000/ 64 <sup>2</sup> , 1000/ 128 <sup>2</sup> , 1000/ 256 <sup>2</sup> , 1000/ 512 <sup>2</sup>
Jumlah iteration	10000	10000

Tabel IV merupakan *hyperparameter* untuk pengujian menggunakan *cost* warna untuk batik no 2 sampai no 5 pada Gambar 3. Nilai 1 untuk sampel yang menggunakan *cost* warna sedangkan Nilai 2 untuk sampel yang tidak menggunakan *cost* warna. Jumlah kluster tersebut ditentukan dari *elbow method* dari setiap batik. Setiap batik memiliki komposisi warna yang berbeda-beda maka dilakukan dengan *elbow method* satu per satu.

TABEL V  
JUMLAH KLUSTER PADA PENGUJIAN PENGGUNAAN COST WARNA UNTUK BATIK YANG LAIN

Nama Hyperparameter	Batik no 2	Batik no 3	Batik no 4	Batik no 5
Jumlah kluster	3	4	3	4

Tabel V merupakan tabel jumlah kluster pada setiap batik yang akan dilakukan pengujian. Jumlah terdapat diperoleh dari *elbow method*. Jumlah batik tersebut berbeda-beda karena memiliki komposisi warna yang berbeda-beda. setiap batik akan dibandingkan sampel yang menggunakan *cost* warna dengan yang tidak menggunakan *cost* warna.

5) *Pengujian dengan gambar konten yang lain*. Pada pengujian ini menggunakan gambar konten seperti pada Gambar 4. Pengujian ini hanya untuk melihat pengaruh dan dampak *cost* warna akan berpengaruh dengan gambar konten atau tidak. Maka pada pengujian ini menggunakan beberapa pola batik saja. Untuk *hyperparameter*-nya akan dilakukan penyesuaian pada setiap percobaan. Pada setiap percobaan itu akan dilihat pada sisi hasil gambar dan sisi *total cost*.

### C. Kualitas Evaluasi

Kualitas evaluasi akan dilakukan dengan meminta pendapat menurut pakar seni pada bidang batik. Hanya saja hasil evaluasi ini akan dipengaruhi banyak faktor. Penilaian *style* sangat subjektif dan sangat kompleks yang melibatkan selera personal. Maka dari itu sangat sulit untuk mendefinisikan *style*. Setiap orang mungkin memiliki selera yang berbeda. Maka untuk evaluasi ini akan lebih fokus

untuk dapat atau tidak dapat gambar hasil disebut sebagai batik.

### D. Hipotesis

Hipotesis dari penelitian ini adalah dapat menghasilkan gambar yang dapat digolongkan batik oleh pakar dan dapat membuat gambar tersebut lebih *solid* dengan menggunakan perhitungan *cost* warna. Hasil penelitian ini akan diuji dengan kualitas dan kuantitas evaluasi untuk menentukan hipotesis ini tercapai atau tidak.

## V. HASIL PENELITIAN

Hasil penelitian akan dibagi 2 bagian. Sub bab tersebut terdiri dari Hasil Evaluasi kuantitas dan kualitas. Cara evaluasi yang sudah dijelaskan pada bab metode penelitian.

### A. Hasil Kuantitas evaluasi

Hasil penelitian akan dibahas untuk setiap kondisi seperti yang sudah dijelaskan pada metode penelitian. Pada setiap kondisi akan dibahas gambar hasil, *histogram* warna, *total cost*, *cost style* dan konten, dan *cost* warna.

1) *Hasil pengujian penggunaan cost warna*. Berikut ini adalah hasil pengujian penggunaan *cost* warna hasil ini akan menampilkan dari kedua sampel yang menggunakan *cost* warna dan yang tidak menggunakan *cost* warna. Setiap sampel akan dibandingkan dari gambar hasil, pergerakan *cost*- nya dan *histogram* warnanya.



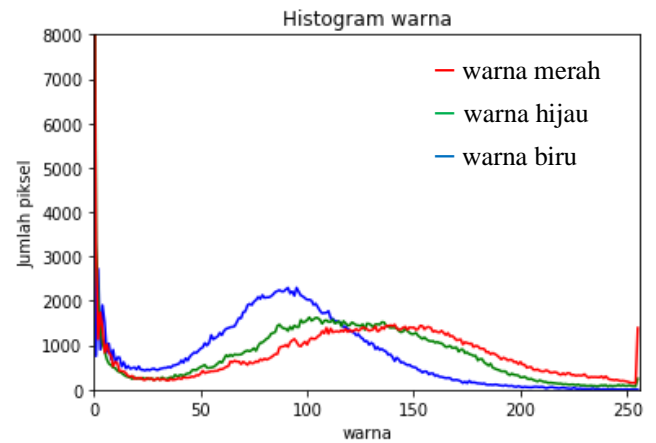
Gambar 6 Hasil gambar dari pengujian dengan penggunaan *cost* warna





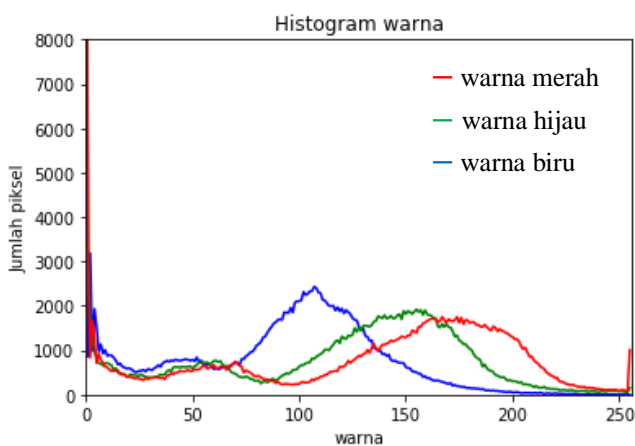
Gambar 7 Hasil gambar dari pengujian dengan tanpa penggunaan *cost* warna

Gambar 6 merupakan gambar hasil dari pengujian dari sampel 1 yang menggunakan *cost* warna sedangkan Gambar 7 merupakan hasil dari pengujian dengan tanpa penggunaan *cost* warna. Dari kedua gambar tersebut ada beberapa perbedaan yang sekilas tidak terlihat. Perbedaan terdapat pada pembentukan pola dengan tanpa *cost* warna lebih banyak pola yang tercipta pada pojok kiri bawah sedangkan pada sampel 1 tidak ada detail pola tercipta pada pojok kiri bawah. Ada 1 perbedaan lagi yaitu warna yang tercipta, perbedaan itu sangat sulit dibandingkan maka akan dibandingkan pada *histogram* warnanya. Tetapi sampel yang menggunakan *cost* warna masih terdapat pola gradasi warna.

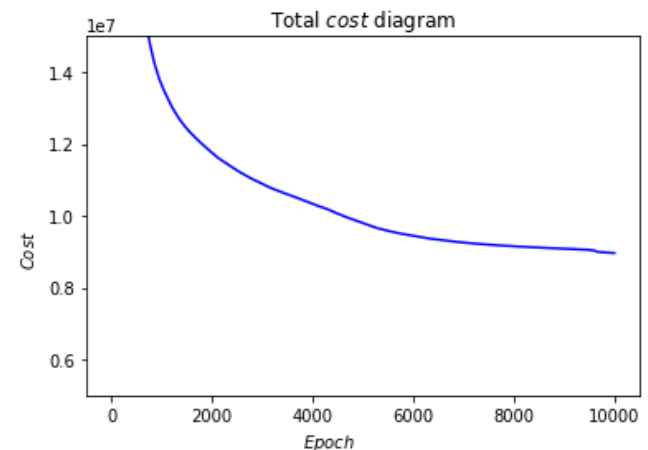


Gambar 9 Histogram warna dari pengujian dengan tanpa penggunaan *cost* warna

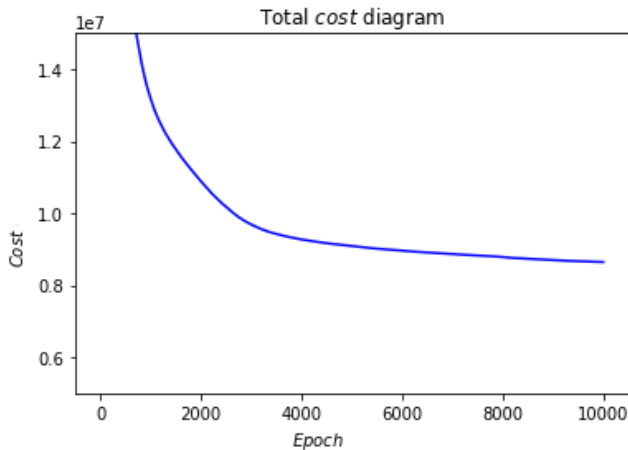
Gambar 8 merupakan histogram warna dari pengujian dengan penggunaan *cost* warna dan Gambar 9 merupakan histogram warna dari pengujian dengan tanpa penggunaan *cost* warna. Dari gambar hasil sulit terlihat perbedaan tetapi dari *histogram* sangat terlihat perbedaan dari ke 2 *histogram* tersebut. Histogram dari sampel 1 yang menggunakan *cost* warna lebih terlihat membentuk kluster warna nya. Kluster itu dapat terlihat menjadi beberapa puncak pada *histogram*. Puncak tersebut sangat terlihat pada warna hijau. Pada Gambar 9, warna hijau dan merah sangat terlihat lebih landai dari pada Gambar 8. Hal tersebut terjadi karena warna piksel pada sampel 1 lebih membentuk kluster dengan begitu kurva nya akan menjadi tidak landai.



Gambar 8 *Histogram* warna dari pengujian dengan penggunaan *cost* warna

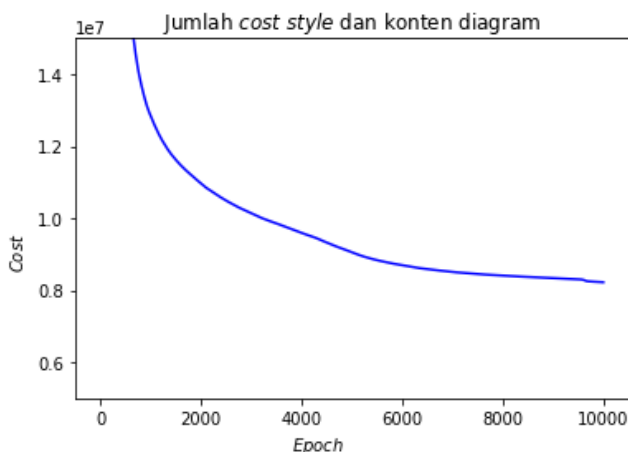


Gambar 10 Total *cost* dari pengujian dengan penggunaan *cost* warna

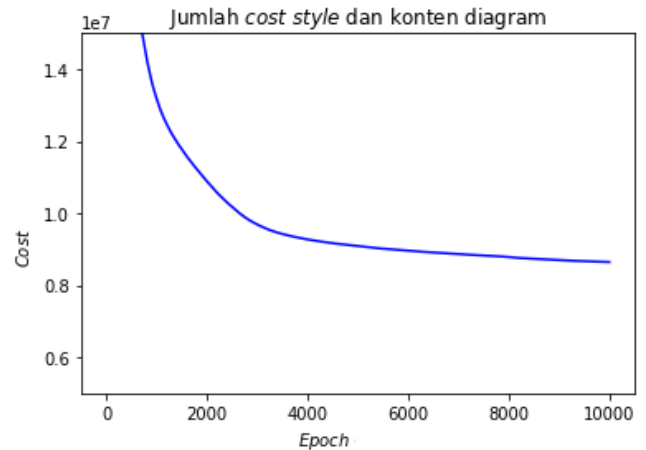


Gambar 11 Total *cost* dari pengujian dengan tanpa penggunaan *cost* warna

Gambar 10 merupakan total *cost* diagram dari pengujian dengan penggunaan *cost* warna sedangkan Gambar 11 merupakan total *cost* diagram dari pengujian dengan tanpa penggunaan *cost* warna. Dari kedua diagram tersebut memiliki *cost* yang besar maka digambarkan dengan skala 1:10.000.000. Dari kedua diagram tersebut menggambarkan penurunan nilai total *cost* dari setiap *epoch* nya. Pada sampel 1 yang menggunakan *cost* warna terlihat penurunan lebih landai pada *epoch* 2000 sampai 5000 dibandingkan dengan tanpa penggunaan *cost* warna. Tetapi pada *epoch* 10000 hanya memiliki perbedaan yang sangat kecil dari ke 2 diagram tersebut. Banyak hal faktor yang dapat menyebabkan hal tersebut. Penurunan tingkat optimasi untuk mungkin terjadi karena terjadi penurunan yang lebih landai pada *epoch* 2000. Mungkin juga terjadi *saddle point* pada data sampel 1 ini. Untuk menunjukkan kemungkinan yang benar akan dilakukan pengujian dengan gambar batik yang lain atau gambar konten yang lain.



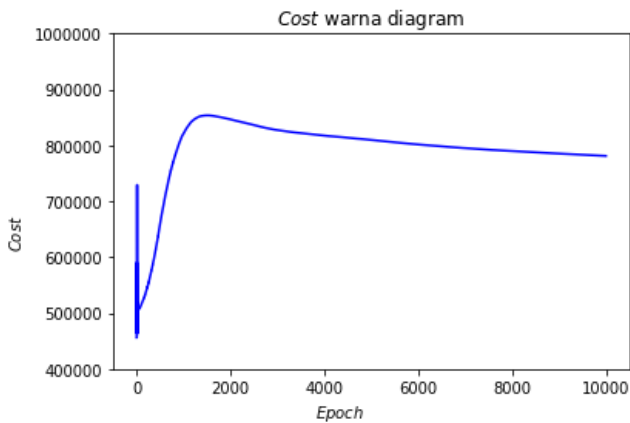
Gambar 12 Jumlah *cost* style dan konten diagram dari pengujian dengan penggunaan *cost* warna



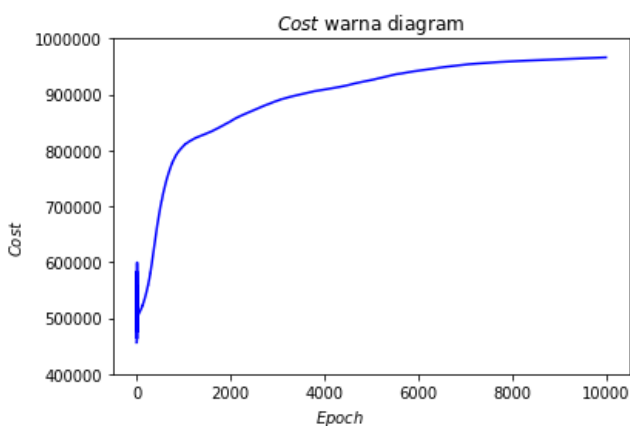
Gambar 13 Jumlah *cost* style dan konten diagram dari pengujian dengan tanpa penggunaan *cost* warna

Gambar 12 merupakan jumlah *cost* style dan konten diagram dari pengujian dengan penggunaan *cost* warna sedangkan pada Gambar 13 merupakan jumlah *cost* style dan konten diagram dari pengujian dengan tanpa penggunaan *cost* warna. Kedua diagram tersebut menggambarkan penurunan *cost* untuk style dan konten sepanjang *epoch*. Jika lebih cepat turun berarti lebih cepat algoritma tersebut membentuk pola style dan pola konten. Pada Gambar 12 yang menggunakan *cost* warna, penurunannya lebih landai terlihat pada *epoch* 2000 sampai *epoch* 5000 dibandingkan dengan diagram yang tidak menggunakan *cost* warna. Sedangkan pada *epoch* 10000, pada Gambar 12 yang menggunakan *cost* memiliki *cost* yang lebih kecil dibandingkan pada *cost* yang tidak menggunakan perhitungan *cost* warna. Banyak hal faktor yang dapat menyebabkan hal tersebut. Penurunan tingkat optimasi untuk mungkin terjadi karena terjadi penurunan yang lebih landai pada *epoch* 2000. Mungkin juga terjadi *saddle point* pada data sampel 1 ini. Untuk menunjukkan kemungkinan yang benar akan dilakukan pengujian dengan gambar batik yang lain atau gambar konten yang lain.





Gambar 14 Cost warna diagram dari pengujian dengan penggunaan cost warna

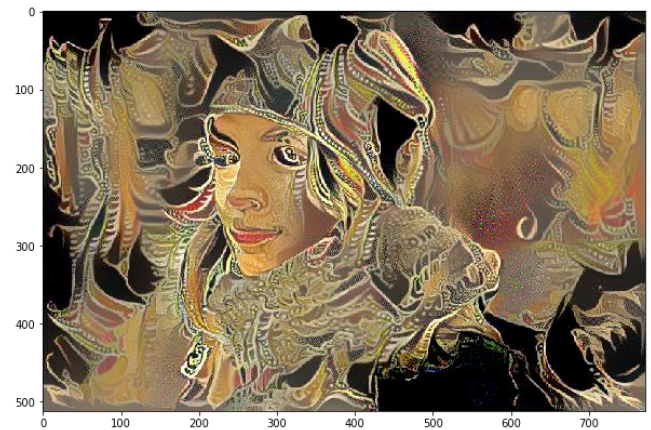


Gambar 15 Cost warna diagram dari pengujian dengan tanpa penggunaan cost warna

Gambar 14 merupakan *cost* warna diagram dari pengujian dengan penggunaan *cost* warna sedangkan Gambar 15 merupakan *cost* warna diagram dari pengujian dengan tanpa penggunaan *cost* warna. Pada sampel yang tidak menggunakan *cost* warna tetapi sampel tersebut tetap dicoba dihitung setiap *epoch*-nya. Jika *cost* warna menurun menandakan *cost* warna semakin kecil. Jika *cost* warna kecil berarti warna yang dihasilkan semakin dekat dengan kluster-kluster yang dihasilkan. Pada Gambar 14 terlihat *cost* warnanya jauh lebih rendah dan terus menurun setiap *epoch* nya dibandingkan dengan Gambar 15. Pada sampel yang tidak memperhitungkan *cost* warna setiap *epoch* nya naik terus. Hal tersebut terjadi karena gambar yang dihasilkan dengan arsitektur yang tidak menggunakan *cost* warna, menghasilkan warna dengan tidak ada batas, sedangkan arsitektur yang menggunakan *cost* warna menghasilkan warna yang mendekati kluster. Jika warna yang dihasilkan menjauhi kluster warna maka akan membuat *cost* lebih besar tetapi hal tersebut mungkin terjadi karena mungkin warna tersebut dapat membuat *cost style* dan konten menjadi jauh lebih kecil.

Pada pengujian ini dapat dilihat warna yang dihasilkan lebih *solid* karena lebih banyak warna-warna yang menjadi kluster-kluster dapat dilihat dari *histogram* warna yang dihasilkan dan dapat dilihat juga dari diagram *cost* warna yang lebih menurun yang menandakan warna – warna yang dihasilkan lebih dekat dengan kluster – kluster warna. Tetapi jika dilihat pada Gambar 6 masih terdapat gradasi warna dari coklat tua ke coklat muda. Hal tersebut dapat terlihat juga pada kurva *histogram* warna nya yang tidak menjadi *peak*. Selain pengaruh terhadap warna, pengaruh terhadap perpindahan pola batik dan konten juga terjadi. Pengaruh tersebut terdapat pada *epoch* 2000 sampai 5000 yang menjadi kurva penurun *cost*- nya menjadi melandai.

2) Hasil pengujian pengaruh peningkatan jumlah kluster warna. Berikut ini adalah hasil pengujian pengaruh peningkatan jumlah kluster warna hasil ini akan menampilkan dari kedua sampel yang memiliki jumlah kluster yang berbeda. Setiap sampel akan dibandingkan dari gambar hasil, pergerakan *cost* nya dan *histogram* warna nya.



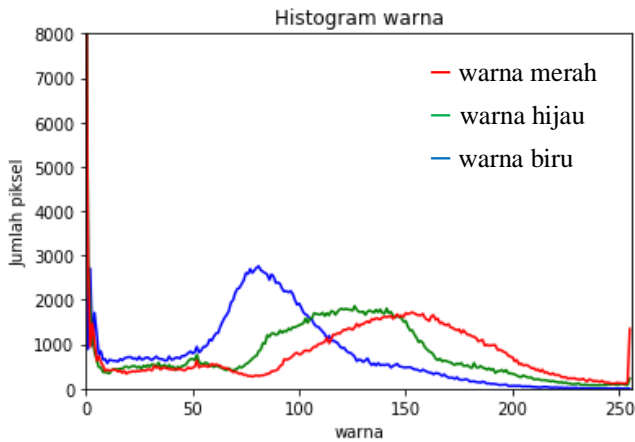
Gambar 16 Hasil gambar dari pengujian dengan jumlah kluster warna sebanyak 3



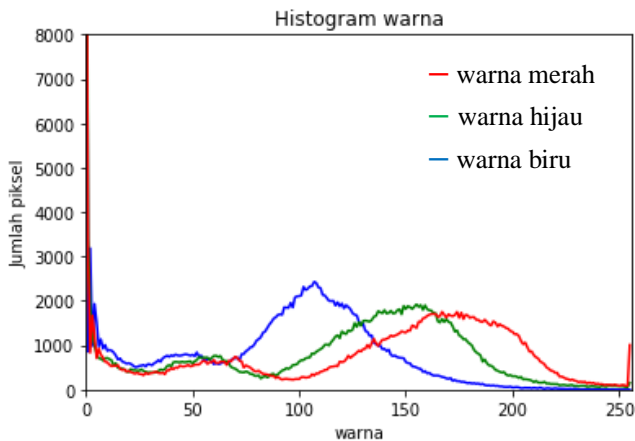
Gambar 17 Hasil gambar dari pengujian dengan jumlah kluster warna sebanyak 2

Gambar 16 merupakan hasil gambar dari pengujian dengan jumlah kluster sebanyak 3 sedangkan Gambar 17

merupakan hasil gambar dari pengujian dengan jumlah kluster sebanyak 2. Sekilas dari kedua gambar tersebut tidak memiliki perbedaan. Dengan melihat pada *histogram* warna akan lebih terlihat perbedaan dari kedua gambar tersebut.

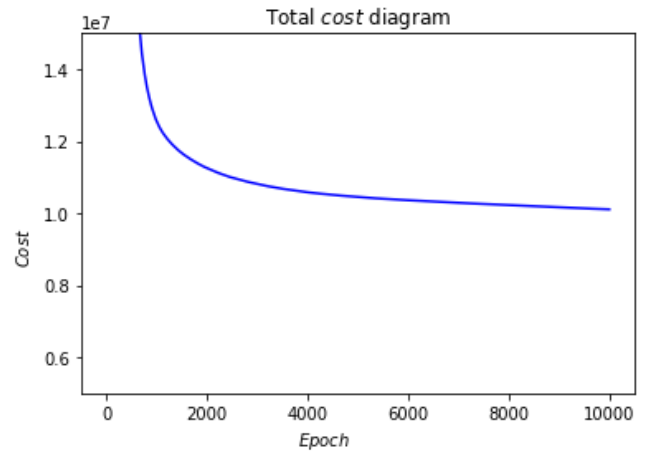


Gambar 18 *Histogram* warna dari pengujian dengan jumlah kluster warna sebanyak 3

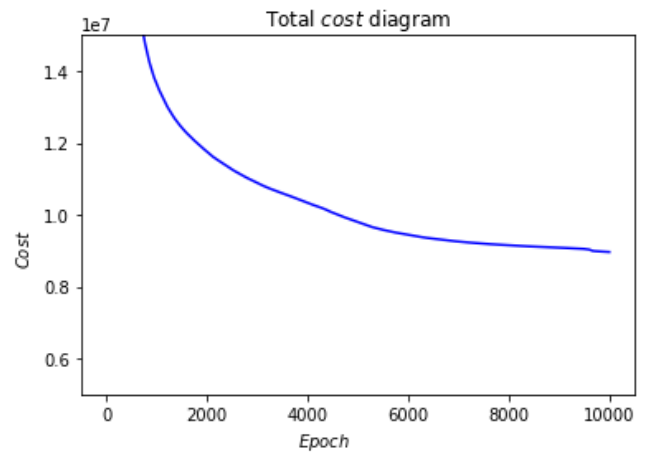


Gambar 19 *Histogram* warna dari pengujian dengan jumlah kluster warna sebanyak 2

Gambar 18 merupakan *histogram* warna dari pengujian dengan jumlah kluster warna sebanyak 3 sedangkan Gambar 19 merupakan *histogram* warna dari pengujian dengan jumlah kluster warna sebanyak 2. Pada kedua *histogram* tersebut tidak terlalu memiliki banyak perbedaan. Pada Gambar 18 terlihat sedikit lebih landai pada kurva warna merah dan hijau. Perbedaan jumlah kluster yang kecil membuat perbedaan yang kecil juga pada *histogram* warnanya. Semakin jauh angka jumlah kluster dari yang disarankan dari *elbow method* maka semakin landai pula kurva pada *histogram* warnanya. Penambahan 1 kluster tersebut dengan perbedaan yang kecil pada *elbow method*, akan memberikan 1 kluster yang dekat dengan kluster yang lain, maka hal tersebut tidak akan memiliki perbedaan. Tetapi jika memberikan jumlah kluster yang besar akan memberikan perbedaan yang lebih besar.

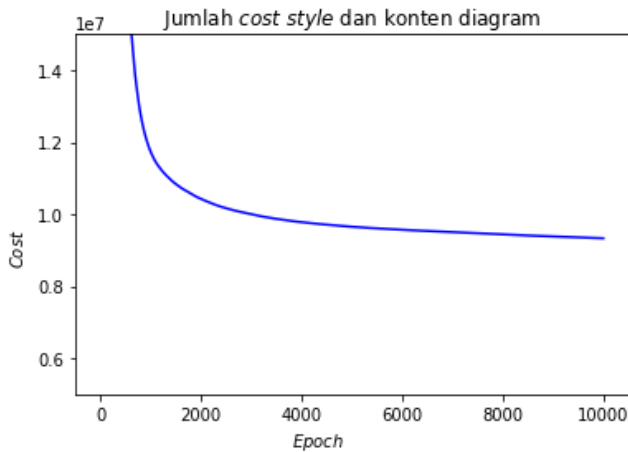


Gambar 20 *Total cost* dari pengujian dengan jumlah kluster warna sebanyak 3

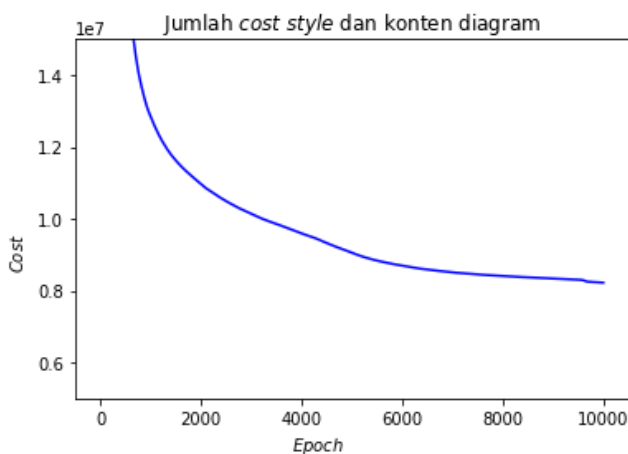


Gambar 21 *Total cost* dari pengujian dengan jumlah kluster warna sebanyak 2

Gambar 20 merupakan diagram *total cost* dari pengujian dengan jumlah kluster warna sebanyak 3 sedangkan Gambar 21 merupakan diagram *total cost* dari pengujian dengan jumlah kluster warna sebanyak 2. Kedua diagram tersebut memiliki beberapa perbedaan. Pada Gambar 20 memiliki kurva yang lebih landai dan pada epoch 10000 mendapatkan *cost* yang lebih tinggi dibandingkan pada Gambar 21. Hal tersebut disebabkan karena menghasilkan warna dengan 2 kluster yang terpisah akan lebih mudah dari pada ada 3 kluster yang mungkin 1 kluster tersebut sangat dekat dengan kluster yang lain. Hal tersebut akan menyulitkan warna tersebut masuk ke kluster yang mana. Tetapi kedua diagram tersebut masih selalu turun. Maka walaupun jumlah kluster berbeda *cost* nya selalu turun tidak membuat *cost* menjadi naik, untuk mendapatkan *total cost* yang paling minimal harus mengikuti jumlah dari *elbow method*.

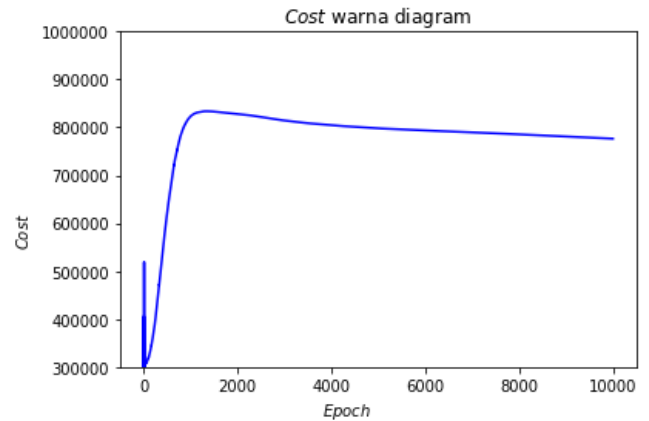


Gambar 22 Jumlah *cost style* dan konten diagram dari pengujian dengan jumlah kluster warna sebanyak 3

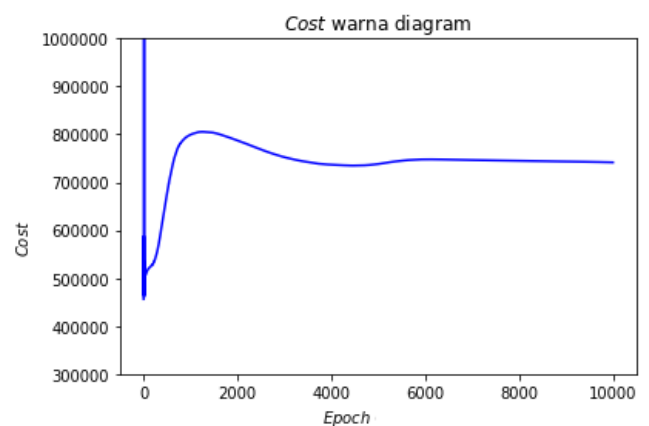


Gambar 23 Jumlah *cost style* dan konten diagram dari pengujian dengan jumlah kluster warna sebanyak 2

Gambar 22 merupakan jumlah *cost style* dan konten diagram dari pengujian dengan jumlah kluster warna sebanyak 3 sedangkan Gambar 23 merupakan jumlah *cost style* dan konten diagram dari pengujian dengan jumlah kluster warna sebanyak 2. Kedua diagram sama dengan penurunan *total cost*. Diagram jumlah *cost style* dan konten pun melandai pada jumlah kluster warna sebanyak 3. Hal tersebut disebabkan karena menghasilkan warna dengan 2 kluster yang terpisah akan lebih mudah dari pada ada 3 kluster, 1 kluster tersebut sangat dekat dengan kluster yang lain. Hal tersebut akan menyulitkan warna tersebut masuk ke kluster yang mana



Gambar 24 *Cost* warna diagram dari pengujian dengan jumlah kluster warna sebanyak 3



Gambar 25 *Cost* warna diagram dari pengujian dengan jumlah kluster warna sebanyak 2

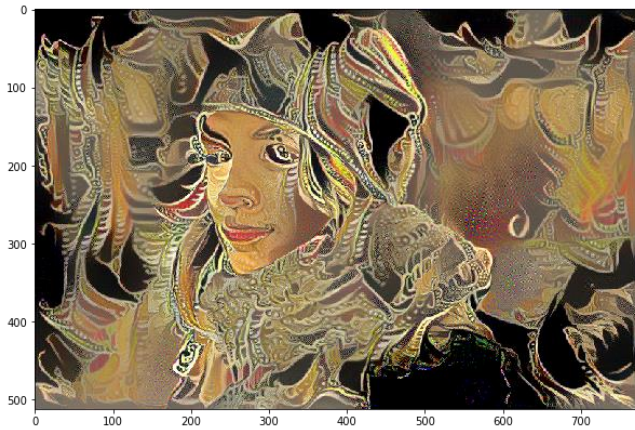
Gambar 24 merupakan *cost* warna diagram dari pengujian dengan jumlah kluster warna sebanyak 3 sedangkan Gambar 25 merupakan *cost* warna diagram dari pengujian dengan jumlah kluster warna sebanyak 2. Kedua diagram tersebut memiliki *cost* yang hampir sama pada *epoch* 10000. Pada Gambar 25 terdapat kenaikan lagi pada *epoch* 4000 sedangkan pada Gambar 5.19. Hal tersebut tidak ada. Perbedaan 1 pada jumlah kluster warna tidak terlalu berdampak pada *cost* warna pada *epoch* 10000.

Pada hasil pengujian dengan jumlah kluster warna, tidak terlalu terlihat perbedaannya jika hanya berbeda pada jumlah 1 kluster tetapi jika jumlah kluster nya jauh berbeda akan semakin terlihat perbedaannya pada *histogram* warnanya. Jika jumlah kluster menggunakan dari hasil *elbow method* maka *histogram* warna lebih memisah menjadi kluster. Jumlah kluster tidak terlalu berdampak terhadap *cost*.

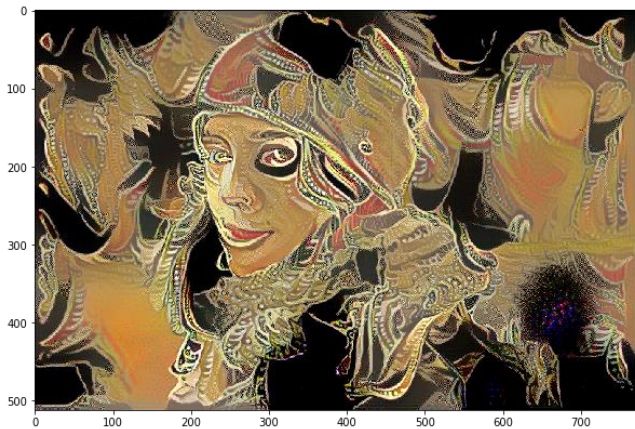
3) Hasil pengujian pengaruh peningkatan *weight factor* warna. Berikut ini adalah hasil pengujian pengaruh peningkatan *weight factor* warna hasil ini akan menampilkan dari ketiga sampel yang memiliki *weight*



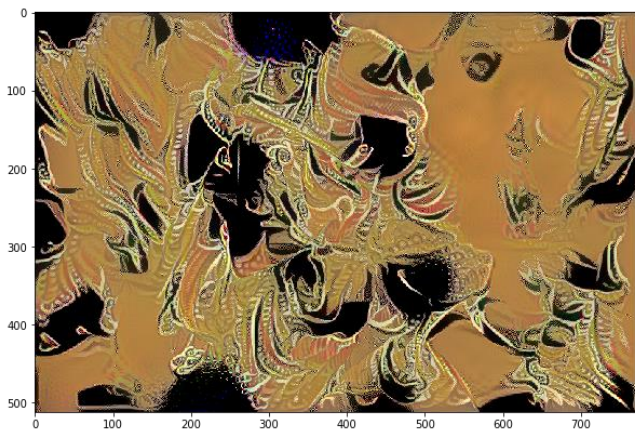
factor yang berbeda. Setiap sampel akan dibandingkan dari gambar hasil, pergerakan *cost* nya dan *histogram* warna nya.



Gambar 26 Hasil gambar dari pengujian dengan *weight factor* warna 50

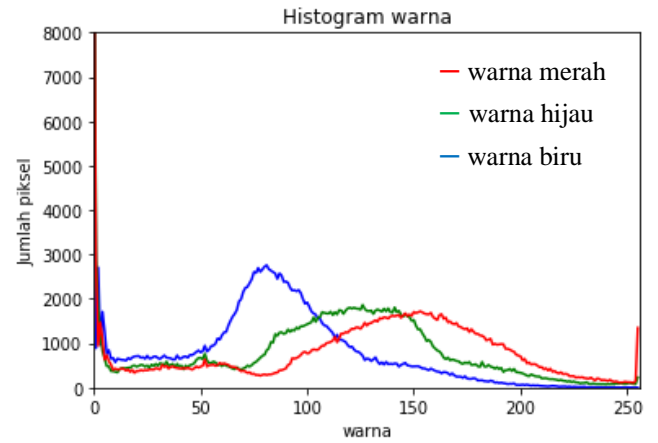


Gambar 27 Hasil gambar dari pengujian dengan *weight factor* warna 100

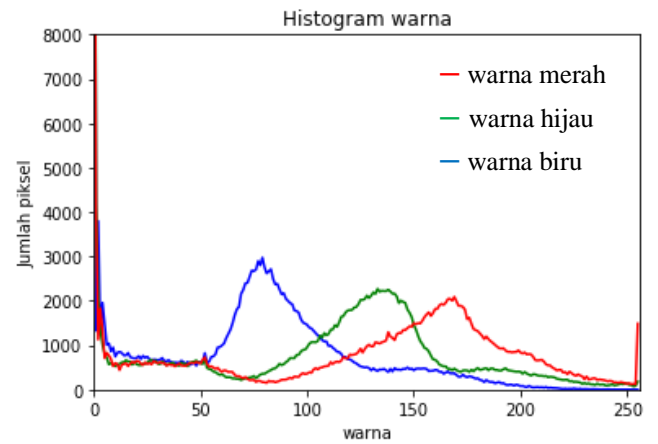


Gambar 28 Hasil gambar dari pengujian dengan *weight factor* warna 1000

Pada Gambar 26 hasil gambar dari pengujian dengan *weight factor* warna 50 sedangkan Gambar 27 hasil gambar dari pengujian dengan *weight factor* warna 100 sedangkan Gambar 28 hasil gambar dari pengujian dengan *weight factor* warna 1000. Pada ketiga gambar tersebut sangat terlihat perbedaannya. Pada Gambar 28 gambar konten sudah tidak tampak karena *weight factor* untuk warna terlalu besar. Perbedaan ini akan lebih terlihat pada *histogram* warnanya.

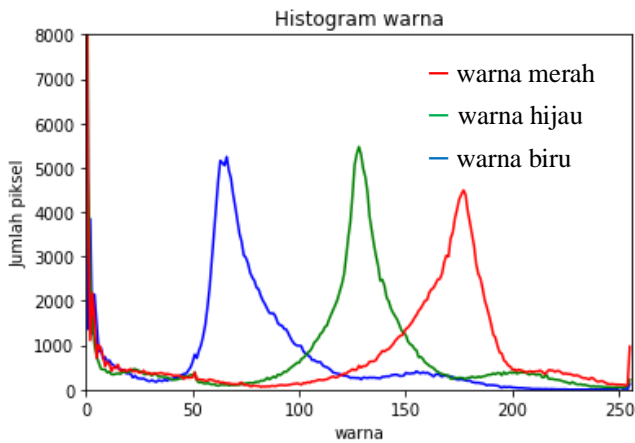


Gambar 29 *Histogram* warna dari pengujian dengan *weight factor* warna 50



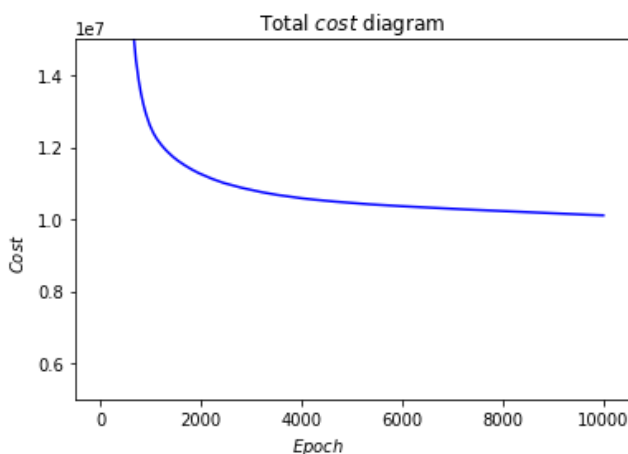
Gambar 30 *Histogram* warna dari pengujian dengan *weight factor* warna 100



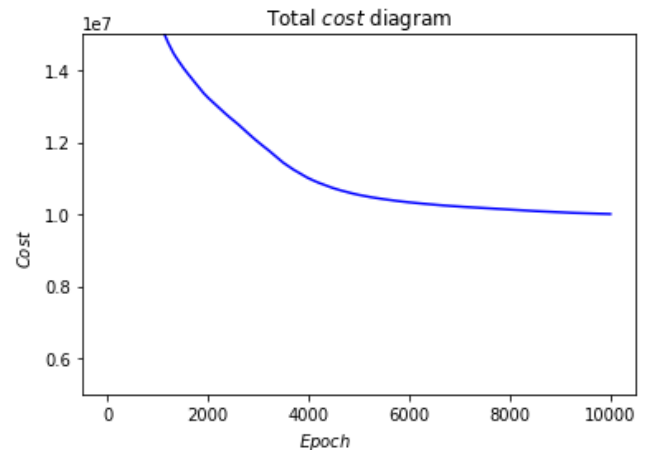


Gambar 31 Histogram warna dari pengujian dengan *weight factor* warna 1000

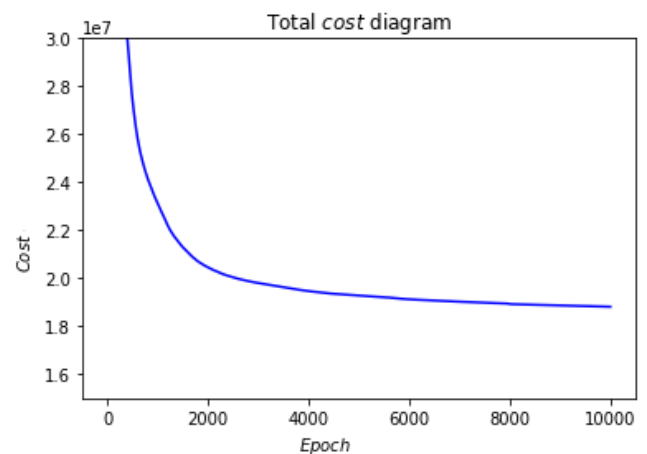
Gambar 29 merupakan *histogram* warna dari pengujian dengan *weight factor* warna 50 sedangkan Gambar 30 merupakan *histogram* warna dari pengujian dengan *weight factor* warna 100 sedangkan Gambar 31 merupakan *histogram* warna dari pengujian dengan *weight factor* warna 1000. Dari ketiga *histogram* tersebut dapat terlihat pergerakan curva nya setiap dinaikan *weight factor* warnanya. Pada Gambar 31 dapat sangat terlihat jika curva setiap warna memuncak sangat tajam. Kurva yang memuncak sangat tajam tersebut menandakan jika warnanya menjadi suatu kluster. Hal tersebut terjadi karena *weight factor* pada sampel 3 sangat besar yaitu 1000, maka setiap ada perubahan kecil yang membuat warna yang dihasilkan menjauh dari kluster akan dampaknya lebih besar pada perhitungan total *cost*. Dampak tersebut yang akan mengganggu proses perpindahan *style* karena perubahan pada perhitungan *cost* warna akan menghasilkan *cost* yang lebih besar dari pada *cost style* dan *cost* konten.



Gambar 32 Total *cost* dari pengujian dengan *weight factor* warna 50

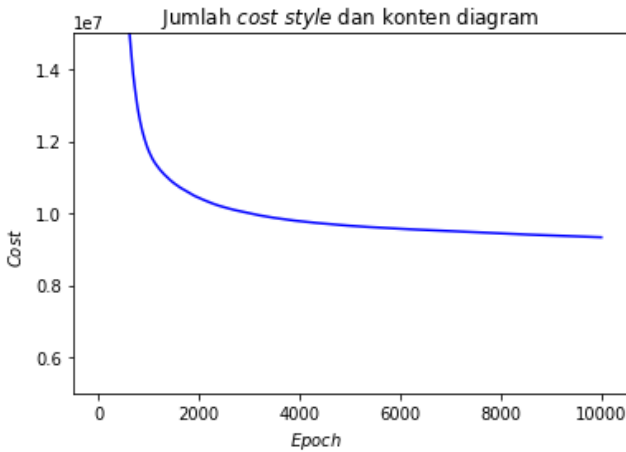


Gambar 33 Total *cost* dari pengujian dengan *weight factor* warna 100

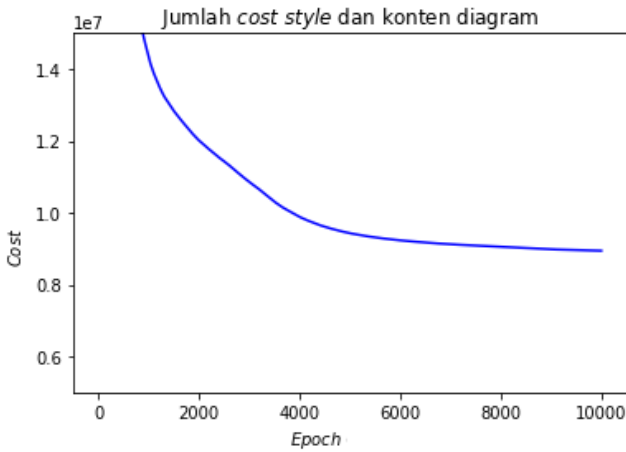


Gambar 34 Total *cost* dari pengujian dengan *weight factor* warna 1000

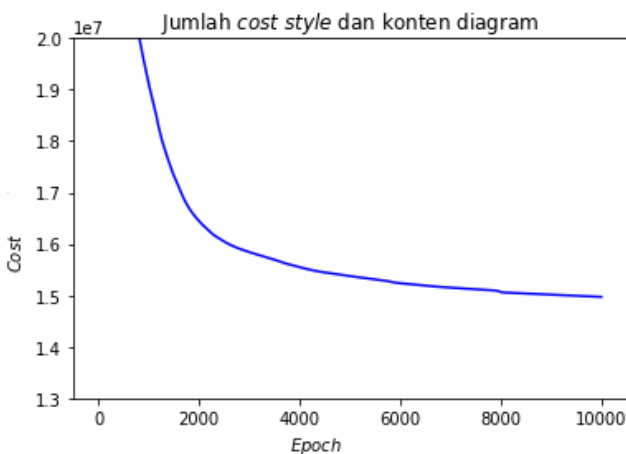
Gambar 32 merupakan Total *cost* dari pengujian dengan *weight factor* warna 50 sedangkan Gambar 33 Total *cost* dari pengujian dengan *weight factor* warna 100 sedangkan Gambar 34 Total *cost* dari pengujian dengan *weight factor* warna 1000. Total *cost* pada Gambar 33 memiliki curva yang sedikit melandai pada *epoch* 2000 sampai 4000. Tetapi pada Gambar 33 memiliki total *cost* yang sedikit lebih kecil dengan Gambar 32 padahal pada sampel 2 memiliki *weight factor* 2x dibandingkan sampel 1 tetapi memiliki *cost* yang sama. Sedangkan Gambar 34 memiliki *cost* yang sangat tinggi karena memiliki *weight factor* yang tinggi juga yaitu 1000. Selain dari *weight factor* tinggi nya total *cost* juga dari proses perpindahan *style* dan konten yang terganggu yang membuat *style* dan konten tidak terbentuk yang membuat *cost*-nya tidak turun.



Gambar 35 Jumlah *cost style* dan konten diagram dari pengujian dengan *weight factor* warna 50

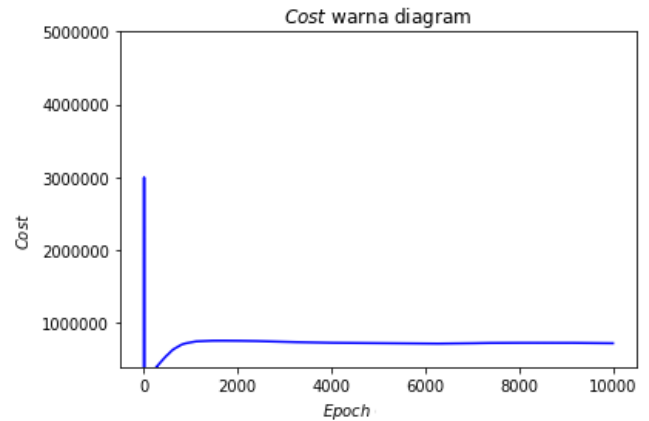


Gambar 36 Jumlah *cost style* dan konten diagram dari pengujian dengan *weight factor* warna 100

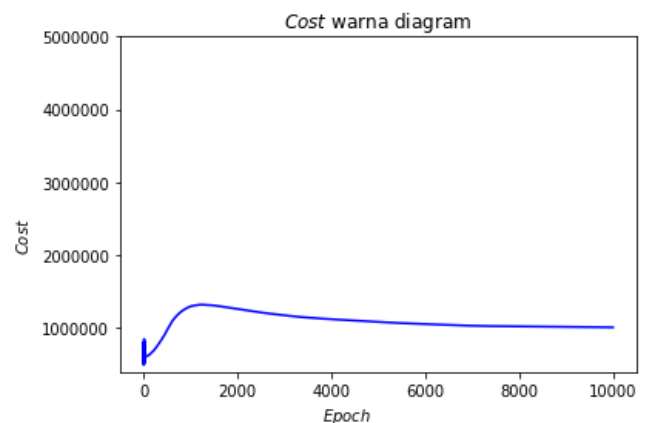


Gambar 37 Jumlah *cost style* dan konten diagram dari pengujian dengan *weight factor* warna 1000

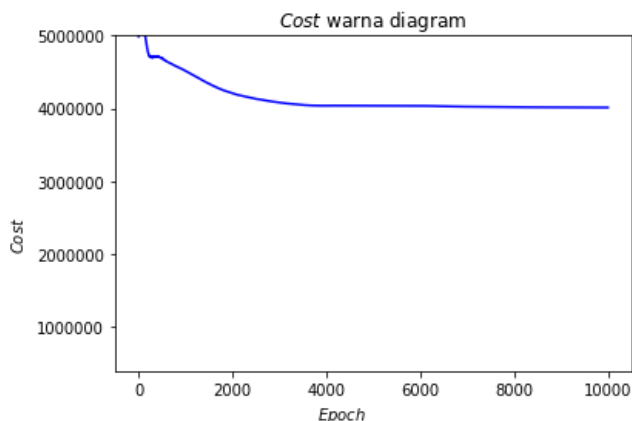
Gambar 35 merupakan jumlah *cost style* dan konten diagram dari pengujian dengan *weight factor* warna 50 sedangkan Gambar 36 merupakan Jumlah *cost style* dan konten diagram dari pengujian dengan *weight factor* warna 100 sedangkan Gambar 37 merupakan Jumlah *cost style* dan konten diagram dari pengujian dengan *weight factor* warna 1000. Pada kurva jumlah *cost style* dan konten juga terjadi bahwa sampel 2 yang memiliki *weight factor* warna 100 dapat *cost* lebih rendah pada *epoch* 10000 dari pada sampel 1 yang memiliki *weight factor* 50 walaupun pada *epoch* 2000 sampai 4000 sampel 2 memiliki jumlah *cost style* dan konten yang lebih tinggi dari pada sampel 1. Sedangkan sampel 3 yang memiliki *weight factor* warna 1000 mendapatkan *cost style* dan konten yang jauh lebih tinggi. Hal tersebut menandakan jika *weight factor* terlalu besar dapat menghambat proses *transfer style*. Maka *style* dan konten tidak dapat tercipta pada gambar yang di hasilkan. Untuk mendapatkan *cost style* dan konten yang paling kecil harus dicari *weight factor* yang paling sesuai juga. Tidak selalu *weight factor* warna sangat kecil maka mendapatkan *cost style* dan konten paling kecil juga.



Gambar 38 *Cost warna diagram* dari pengujian dengan *weight factor* warna 50



Gambar 39 *Cost warna diagram* dari pengujian dengan *weight factor* warna 100

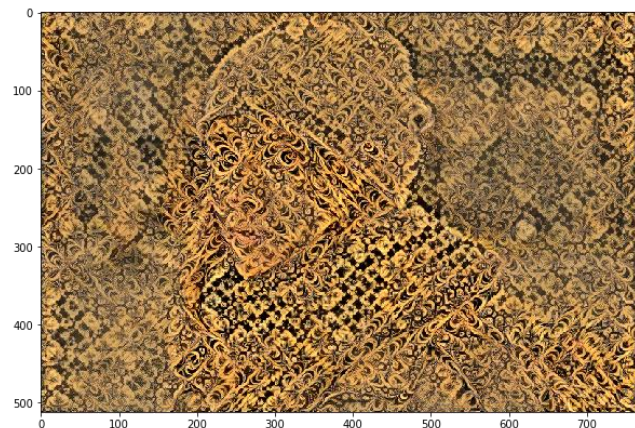


Gambar 40 Cost warna diagram dari pengujian dengan *weight factor* warna 1000

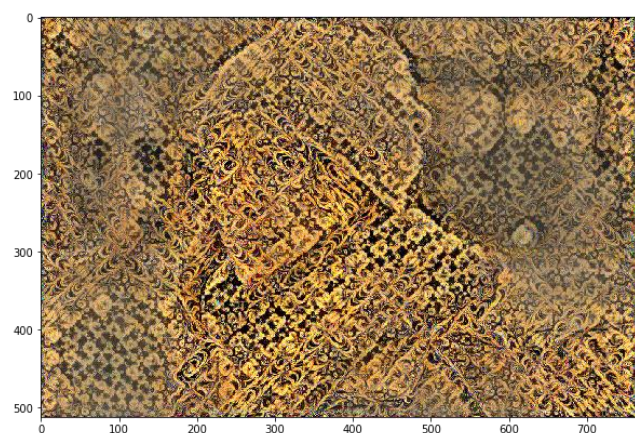
Gambar 38 merupakan *cost* warna diagram dari pengujian dengan *weight factor* warna 50 sedangkan Gambar 39 merupakan *Cost* warna diagram dari pengujian dengan *weight factor* warna 100 sedangkan Gambar 40 merupakan *Cost* warna diagram dari pengujian dengan *weight factor* warna 1000. Dari ketiga diagram tersebut semua menggambarkan jika lebih besar *weight factor* warna nya maka akan mendapatkan *cost* warna yang lebih besar. Tetapi jika kita hitung *cost*-nya diluar *weight factor* maka akan mendapatkan *cost* yang lebih kecil. Pada Gambar 38 mendapatkan sekitar 800000 pada epoch 10000 sedangkan pada Gambar 39 mendapatkan sekitar 1100000 pada epoch 10000. Dari kedua sampel tersebut yang memiliki 2x lebih besar *weight* faktornya tetapi *cost* nya tidak lebih besar 2x lipat. Maka sampel dengan *weight factor* warna 100 memiliki *cost* yang lebih kecil. Hal tersebut terjadi karena proses training akan lebih fokus untuk menurunkan *cost* warna dari pada *cost style* dan *cost* konten karena *cost* warna memiliki bobot yang lebih besar dari pada *cost* yang lain. Perubahan kecil yang membuat warna lebih jauh dari kluster akan membuat *cost* menjadi lebih jauh lebih besar.

Pada pengujian pengaruh peningkatan jumlah kluster warna dapat disimpulkan beberapa hal. Jika *weight factor* terlalu besar maka akan dapat mengganggu ke proses *transfer style* dan konten pada gambar hasil. Untuk mendapatkan total *cost* terendah, harus menggunakan *weight factor* warna yang tepat. Tidak selalu *weight factor* warna makin kecil makin kecil juga total *cost* nya. *Weight factor* warna sangat berpengaruh pada warna yang dihasilkan dapat dilihat pada histogram yang dihasilkan.

1) Hasil pengujian dengan pola batik yang lain. Pada hasil pengujian dengan pola batik akan dibahas setiap batik akan dibahas dengan membandingkan *cost style*, *cost* konten, *histogram* warna, dan *cost* total. Perbandingan telah dilakukan dengan 2 sampel. Satu sampel tidak menggunakan fungsi perhitungan *cost* warna dengan yang lainnya menggunakan fungsi perhitungan *cost* warna. Setiap batik akan memiliki 2 sampel tersebut.

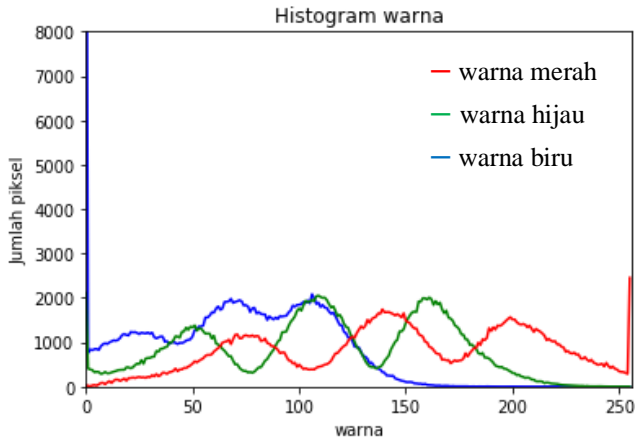


Gambar 41 Hasil gambar dari pengujian dengan penggunaan *cost* warna pada batik no 2

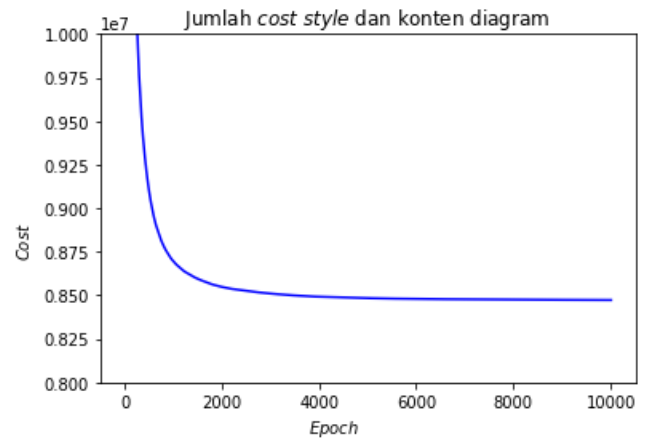


Gambar 42 Hasil gambar dari pengujian dengan tanpa penggunaan *cost* warna pada batik no 2

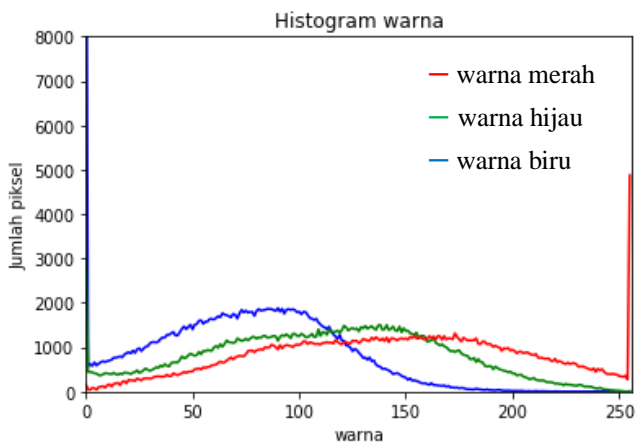
Gambar 41 merupakan gambar hasil dari pengujian dengan penggunaan *cost* warna pada batik no 2 sedangkan Gambar 42 merupakan gambar hasil dari pengujian dengan tanpa penggunaan *cost* warna pada batik no 2. Pada kedua gambar hasil tersebut sangat sulit untuk menemukan perbedaannya. Dari kedua gambar hasil tersebut memiliki pola gambar yang sangat sama dan memiliki perbedaan sedikit warna. Perbedaan warna dapat dilihat lebih lanjut pada kedua *histogram* warna.



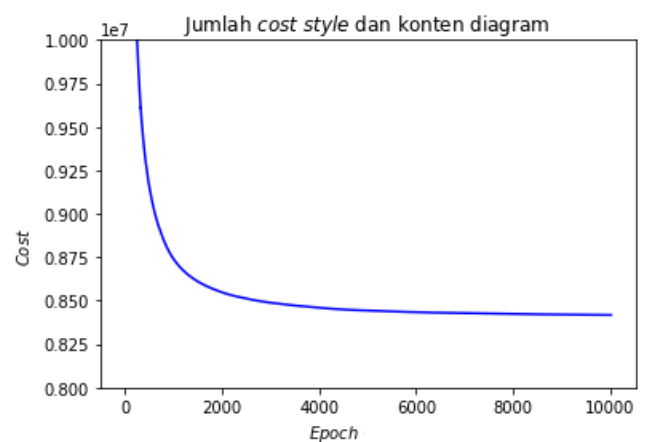
Gambar 43 Histogram warna dari pengujian dengan penggunaan *cost* warna pada batik no 2



Gambar 45 Jumlah *cost style* dan konten diagram dari pengujian dengan penggunaan *cost* warna pada batik no 2



Gambar 44 Histogram warna dari pengujian dengan tanpa penggunaan *cost* warna pada batik no 2

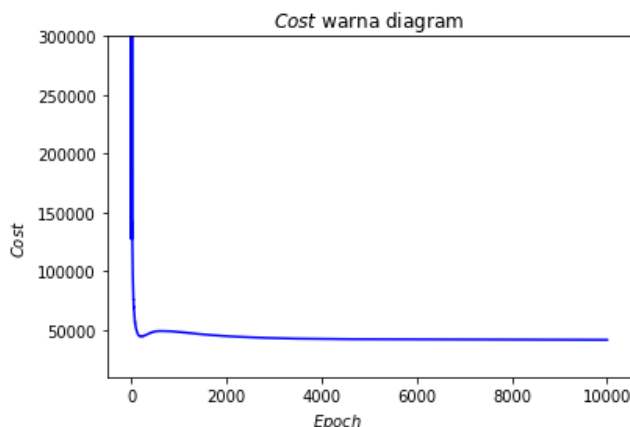


Gambar 46 Jumlah *cost style* dan konten diagram dari pengujian dengan tanpa penggunaan *cost* warna pada batik no 2

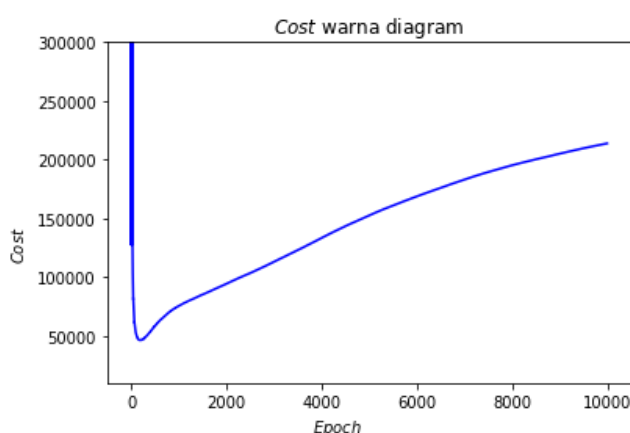
Gambar 43 merupakan *histogram* warna dari pengujian dengan penggunaan *cost* warna pada batik no 2 sedangkan Gambar 44 merupakan *histiogram* warna dari pengujian dengan tanpa penggunaan *cost* warna pada batik no 2. Dari kedua *histogram* memiliki perbedaan yang sangat terlihat. Pada Gambar 43 yang menggunakan perhitungan *cost* warna, memiliki kurva yang memiliki beberapa puncak. Hal tersebut membutuhkan ada nya kluster pada warna-warna tersebut. Sedangkan pada gambar Gambar 44 yang mempunyai kurva yang landai. Hal tersebut membutuhkan warna tersebut tersebar merata. Hal tersebut terjadi karena warna piksel pada sampel 1 lebih membentuk kluster dengan begitu kurva nya akan menjadi tidak landai.

Gambar 45 merupakan diagram jumlah *cost style* dan konten diagram dari pengujian dengan penggunaan *cost* warna pada batik no 2 sedangkan Gambar 46 merupakan diagram jumlah *cost style* dan konten diagram dari pengujian dengan tanpa penggunaan *cost* warna pada batik no 2. Dari kedua diagram jumlah *cost style* dan konten tersebut sangat sedikit perbedaannya. Dari data *cost* nya hanya memiliki perbedaan yang sangat kecil dari kedua sampel tersebut pada *epoch* 10000. Terlihat juga dari gambar hasil dari kedua gambar hasil tidak memiliki perbedaan pola gambar dan *style*. Hal tersebut membuktikan *cost* warna tidak mengganggu perpindahan *style* pada batik no 2. Hal tersebut terjadi karena perhitungan *cost* warna tidak mengganggu perpindahan *style* dan konten. Maka *cost style* dan konten dapat turun seperti tidak menggunakan perhitungan *cost* warna.





Gambar 47 Cost warna diagram dari pengujian dengan penggunaan cost warna pada batik no 2



Gambar 48 Cost warna diagram dari pengujian dengan tanpa penggunaan cost warna pada batik no 2

Gambar 47 merupakan *cost* warna diagram dari pengujian dengan penggunaan *cost* warna pada batik no 2 sedangkan Gambar 48 merupakan *cost* warna diagram dari pengujian dengan tanpa penggunaan *cost* warna pada batik no 2. Dari diagram tersebut terlihat jelas perbedaannya. Pada sampel yang menggunakan *cost* warna, *cost* warna akan turun terus sampai sekitar 60000. Sedangkan sampel yang tidak menggunakan *cost* warna *cost* warna naik terus. Pada *epoch* 10000 hampir *cost* warna menjadi 250000.

Pada pengujian dengan menggunakan batik no 2, *cost style* dan konten memiliki angka yang hampir sama antara sampel yang menggunakan *cost* warna dengan sampel yang tidak menggunakan *cost* warna. Hal tersebut tidak terjadi pada pengujian pada batik no 1. Pada pengujian pada batik no 1, memiliki perbedaan pada *epoch* 2000 sampai 5000. Hal tersebut tidak terjadi pada pengujian pada batik no 2. Hal tersebut dimungkinkan terjadi karena pola yang akan dipindahkan berbeda. Pada pola batik no 2, penggunaan *cost*

warna tidak menghambat perpindahan pola dan konten pada *epoch* 2000 sampai 5000.

Pada pengujian dengan menggunakan batik no 3 sampai no 5 dilakukan dengan cara yang sama dengan pengujian dengan menggunakan batik no 2. Hasil dari pengujian batik no 3 sampai no 5 menghasilkan hasil pengamatan yang serupa dengan pengujian no 2. Hasil pengamatan tersebut terkait *histrogram* warna, *cost* warna, *cost style* dan konten, dan *cost total*.

Kesimpulan untuk pengujian pada setiap batik memiliki beberapa butir. Setiap batik yang diuji dengan menggunakan perhitungan *cost* warna, memiliki *histrogram* warna yang menunjukkan adanya kluster warna. Setiap batik yang diuji menunjukkan *cost* warna tidak mengganggu perpindahan *style* dengan *hyperparameter* yang sudah dibahas di bab sebelumnya. Setiap batik yang diuji menggunakan menggunakan perhitungan *cost* warna, memiliki *cost* warna yang cenderung menurun sedangkan batik yang diuji dengan tidak menggunakan *cost* warna, memiliki *cost* warna yang cenderung naik.

5) Hasil pengujian dengan gambar konten yang lain. Hasil pengujian dengan gambar konten yang lain telah dilakukan pada kelima batik dari setiap pengujian telah dibandingkan dari *cost* warna, *cost style* dan konten dan *histrogram* warna. Dari semua pengujian menghasilkan pengamatan yang sama seperti menggunakan Gambar 4. Seluruh pengujian menghasilkan *cost* warna diagram yang menurun. *Cost* warna akan menunjukkan warna yang dihasilkan akan membentuk kluster-kluster agar warna yang dihasilkan lebih *solid*. Dapat diindikasikan dengan curva yang lebih memuncak pada *histrogram* warna.

## B. Hasil Kualitas evaluasi

Hasil dari kualitas evaluasi adalah berupa hasil dari wawancara pakar pada batik tentang hasil batik dari *Neural Transfer Style* dan bagaimana kualitasnya. Menurut pakar sangat sulit untuk menentukan kualitas sebuah batik karena tidak ada patokannya. Biasanya batik yang dihargai mahal dikarenakan dibuat *handmade*, halus dan sangat rapih biasanya pengerjaannya memakan waktu yang sangat lama hingga 1 sampai 6 tahun untuk selebar batik. Banyak sekali peminat untuk batik yang dikerjakan selama 3 tahun yang dapat mencapai harga 150 -250 juta rupiah. Biasanya batik juga mengandung cerita dibalik gambar batik tersebut. Maka sangat sulit menilai kualitas batik hasil dari algoritma *Neural Transfer Style* karena gambar yang dihasilkan banyak faktor yang mempengaruhi. Faktor-faktor tersebut tergantung pada jenis pola batik, gambar konten dan faktor *random* pada pehasilan gambar pada proses inialisasi.

Menurut pakar gambar yang dihasilkan sudah dapat di kategorikan sebagai batik. Gambar tersebut termasuk karya seni batik kreatif. Unsur-unsur yang dipertahankan dalam batik adanya potongan potongan motif batik yang

tampaknya dibuat dengan teknik kolase dan dipadukan dengan digital dalam pengerjaannya.

## VI. KESIMPULAN

*Neural Transfer Style* yang menggunakan perhitungan *cost* warna dapat membuat warna lebih *solid* (warna menjadi kluster-kluster) dapat terlihat pada *histogram* yang dihasilkan. Tetapi masih terdapat pola gradasi pada gambar yang dihasilkan. Perhitungan dengan *cost* warna tidak akan mengganggu proses perpindahan pola dan konten jika *weight faktor* warna dan jumlah kluster warna yang digunakan tepat besarnya. Karena jika terlalu besar maka proses perpindahan *style* dan konten dapat terganggu.

*Neural Transfer Style* dengan menggunakan perhitungan *cost* warna dapat menghasilkan gambar batik yang tergolong batik kreatif oleh pakar. Tetapi untuk kualitas batik tersebut tergantung pada pola batik yang digunakan dan konten yang digunakan. Menurut pakar, batik yang dihargai tinggi adalah batik yang dibuat dengan *handmade*, rapih, dan pembuatannya memakan waktu.

Ada beberapa hal yang dapat dilakukan untuk penelitian berikutnya. Penelitian berikutnya dapat menggunakan arsitektur CNN yang berbeda untuk menghasilkan warna yang lebih *solid*. Ada kemungkinan terdapat 1 cara untuk membuat warna yang dihasilkan lebih *solid* dari pada dengan menggunakan *cost* warna terhadap kluster warna. Penelitian berikutnya dapat menggunakan *hardware* yang lebih baik dari yang digunakan pada penelitian ini. Setiap eksperimen pada penelitian ini dijalankan menggunakan GPU NVIDIA GTX 1060 dan memakan waktu kira-kira 90 menit untuk setiap 10000 *epoch*.

## DAFTAR PUSTAKA

- [1] R. Shamasundari. (2017) Celebrating Indonesia's cultural heritage, batik. [Online]. Tersedia: <https://theaseanpost.com/article/celebrating-indonesias-cultural-heritage-batik>.
- [2] L. A. Gatys, A. S. Ecker and M. Bethge, "Image Style Transfer Using Convolutional Neural Networks," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 2016, pp. 2414-2423.
- [3] H. Nugroho. (2020) Pengertian Motif Batik dan Filosofinya. Kementerian Perindustrian Republik Indonesia.[Online]. Tersedia: [https://bbkb.kemenperin.go.id/index.php/post/read/pengertian\\_motif\\_batik\\_dan\\_filosofinya\\_0](https://bbkb.kemenperin.go.id/index.php/post/read/pengertian_motif_batik_dan_filosofinya_0).
- [4] K. Gurney, *An Introduction to Neural Networks*, London: Routledge, 1997
- [5] I. Goodfellow, Y. Bengio & A. Courville, *Deep Learning*, Cambridge, Massachusetts: The MIT Press, 2016
- [6] S. Albawi, T. A. Mohammed & S. Al-Zawi, "Understanding of a convolutional neural network," *International Conference on Engineering and Technology (ICET)*, Antalya, 2017, pp. 1-6
- [7] K. Yoon, "Convolutional Neural Networks for Sentence Classification," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, D14-1181, pp. 1746-1751.
- [8] K. Simonyan & A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *International Conference on Learning Representations, 2015*, CoRR abs/1409.1556.
- [9] Neurohive. (2018) VGG16 – Convolutional Network for Classification and Detection. [Online]. Tersedia: <https://neurohive.io/en/popular-networks/vgg16/>.
- [10] W. Junjie, *Advances in K-means Clustering, A Data Mining Thinking*, Berlin Heidelberg, Germany: Springer-Verlag, 2012
- [11] C. Yuan & H. Yang, "Research on K-Value Selection Method of K-Means Clustering Algorithm," *J – Multidisciplinary Scientific Journal*. vol. 2, pp. 226-235, 2019
- [12] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85-117, 2015.
- [13] J. Yongcheng, Y. Yezhou, F. Zunlei, Y. Jingwen, Y. Yizhou and S. Mingli, "Neural Style Transfer: A Review," *IEEE Transactions on Visualization & Computer Graphics*, vol. 01, PrePrints pp. 1-1, 2019.
- [14] J. Johnson, A. Alahi and L. Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," *Computer Vision – ECCV 2016*, pp. 694-711, 2016.
- [15] R. Fisher. (2020) CVonline: Image Databases. [Online]. Tersedia: <http://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htm>