

Penerapan CNN dengan Filter Gabor sebagai *feature extractor* untuk *Content-Based Image Retrieval*

Bagas Muharom Hanugrah Hidayat¹, Ricky Eka Putra²

^{1,2} Teknik Informatika, Universitas Negeri Surabaya

¹bagashidayat@mhs.unesa.ac.id

²rickyputra@unesa.ac.id

Abstrak— Seiring dengan perkembangan teknologi informasi, kebutuhan dalam pencarian informasi menjadi hal yang penting. Jika pencarian informasi selama ini dilakukan pada data berjenis teks, maka pada perkembangan teknologi saat ini, memungkinkan adanya pencarian informasi dalam bentuk citra digital. Hal tersebut terjadi karena adanya peningkatan jumlah pustaka digital dalam bentuk citra. Sebuah metode pengembalian citra menjadi komponen utama untuk memecahkan masalah tersebut. CBIR merupakan sistem pengembalian citra yang akan membantu dalam proses pencarian citra dengan memanfaatkan fitur-fiturnya. Penggunaan ekstraksi fitur yang tepat diperlukan untuk mendapatkan fitur tersebut. Pemilihan ekstraksi fitur akan sangat memengaruhi hasil dari CBIR. Salah satu metode yang dapat melakukan ekstraksi fitur pada citra adalah CNN. Metode yang masih dalam satu jenis dalam *deep learning* ini mampu mempelajari fitur citra untuk dimanfaatkan ke dalam bidang visi komputer. Karena itu, CNN menjadi perhatian menarik dalam penelitian ini untuk melakukan CBIR. Penggunaan filter Gabor yang mampu mendapatkan tekstur citra dengan baik juga akan diimplementasikan sebagai filter pada lapisan konvolusi CNN. Dengan menggunakan CNN dan filter gabor, penelitian ini mampu mendapatkan nilai mAP sebesar 0,895 terhadap data uji dengan dataset GHIM10k. Penelitian ini juga membandingkan beberapa metode pengukuran jarak untuk mendapatkan sistem CBIR terbaik.

Kata Kunci— Content Based Image Retrieval, Convolutional Neural Networks, pengukuran jarak, filter Gabor, visi komputer

I. PENDAHULUAN

Saat ini, perkembangan koleksi citra digital sangat berkembang pesat dan belangsung secara kontinu di masa mendatang [1]. Perkembangan jumlah citra digital ini berbanding lurus dengan penggunaan internet yang saat ini juga bergantung pada kesediaan citra digital. Jika umumnya informasi tersedia dalam bentuk teks, maka saat ini informasi juga tersedia dalam bentuk citra digital. Akibatnya, kesediaan informasi pada sebuah citra juga meningkat. Seiring dengan berkembangnya informasi dalam bentuk citra digital ini pencarian informasi dalam bentuk citra digital menjadi masalah yang baru. Dari alasan tersebut maka, dibutuhkan sebuah metode dalam melakukan pencarian citra dengan memanfaatkan citra sebagai data masukan.

Image retrieval merupakan suatu bidang yang membahas mengenai pengembalian beberapa citra dari sebuah koleksi atau dataset citra yang dianggap memiliki kesamaan terhadap suatu citra kueri. Dalam beberapa tahun belakangan, pengimplementasian aplikasi *image retrieval* menjadi bahan menarik sebagai bahan penelitian. Sejauh ini, *image retrieval* dilakukan dengan memanfaatkan fitur dari citra yang akan diolah. Fitur ini diambil melalui tahap ekstraksi menggunakan metode lain. Beberapa ekstraksi fitur yang sering digunakan adalah bentuk, warna, tekstur, garis, dan sebagainya [2]. Fitur-fitur tersebut merupakan fitur dengan level yang rendah. Hasil dari *image retrieval* sendiri sangat bergantung pada fitur-fitur yang dihasilkan pada citra. Semakin baik fitur tersebut, maka semakin baik pula hasil dari *image retrieval*. Proses pengembalian citra dengan memanfaatkan konten visual melalui fitur-fiturnya lebih umum dikenal dengan istilah *Content-Based Image Retrieval*. Namun, istilah “*content-based*” sendiri masih menjadi kendala dalam bidang *image retrieval*. Sebab itu, untuk mengekstrak konten visual dari sebuah citra dibutuhkan adanya metode yang mampu menyamai persepsi manusia.

Belakangan ini, penelitian mengenai *deep learning* sangat ramai dilakukan. Umumnya, penelitian mengenai *deep learning* dilakukan dengan data citra sebagai bahan penelitian. Kajian *deep learning* dalam bidang analisis citra erat kaitannya dengan mempelajari fitur pada citra. Salah satu metode *deep learning* yang menggunakan citra sebagai objeknya adalah CNN atau *Convolutional Neural Network*. CNN telah menarik banyak perhatian dalam bidang visi komputer, salah satunya pada klasifikasi citra [3]. Penelitian pada [4] memanfaatkan teknik *unsupervised CNN* untuk melakukan ekstraksi fitur. Ekstraksi fitur ini sendiri sangat erat kaitannya dengan bidang *image retrieval*. Pemanfaatan ekstraksi fitur dengan metode CNN membuat beberapa penelitian berupaya menerapkan CNN dalam bidang *content-based image retrieval*.

Penelitian mengenai CBIR sudah banyak dilakukan sebelumnya. Seperti pada [1], CBIR menggunakan fitur yang didapat melalui metode HSV, BSIF, CEDD, dan Color Moments. Penggunaan beberapa metode pengukuran jarak juga menjadi perhatian pada penelitian tersebut. Penelitian [5] menggunakan fitur yang lebih kompleks dengan mendapatkan fitur visual primer untuk mendapatkan *bar-shaped structure*

dan *saliency map*. Selain itu, penelitian [6] menerapkan penggunaan filter gabor dalam bidang CBIR. Filter Gabor juga berperan pada [5] untuk mendapatkan tekstur citra. Pada [2], CBIR dilakukan dengan mengombinasikan fitur citra dan melakukan pengambilan teks yang terdapat didalam citra. *Image retrieval* juga dapat dilakuakn dengan melakukan pendekatan *Bag-of-Words* [7]. Pemilihan metode ekstraksi fitur pada bidang CBIR ini masih menjadi tantangan tersendiri. Hingga kini, masih banyak penelitian mengenai CBIR dengan melakukan percobaan pada berbagai ekstraksi fitur.

Secara khusus, CNN adalah metode yang berkaitan dengan citra. Seperti pada [8], metode CNN mendapatkan hasil yang baik untuk klasifikasi pada dataset MNIST Digit. CNN akan berusaha untuk memetakan citra ke dalam kelasnya masing-masing dengan memanfaatkan metode konvolusi untuk mendapatkan fiturnya. Dari fitur tersebut, *fully connected layer* akan melakukan tahap klasifikasi. Penelitian [9] melakukan pelatihan CNN dengan menggunakan *dropout* untuk mengurangi efek *overfitting*. Sementara, itu, penggunaan *Batch Normalization* pada [10] mampu mengurangi masalah seperti *internal covariate shift*.

Penelitian [3] menggunakan metode CNN dalam bidang CBIR dengan menggunakan istilah *Convolutional Neural Codes*. Lapisan dari *fully connected layer* digunakan dengan menafaatkan *softmax* sebagai fungsi aktivasi dan berfungsi untuk memetakan citra pada suatu kelas. Penelitian tersebut juga memanfaatkan penggunaan *multi-scale convolutional neural codes* dengan melakukan percabangan pada lapisan konvolusi. Beberapa metode tambahan seperti *binarization*, *PCA*, dan *random noise* juga diimplementasikan. Istilah *neural codes* ini ditemukan kembali pada [11] dengan memanfaatkan metode *dimensionality reduction* untuk mengurangi dimensi fitur. Penelitian [3] juga memilih metode PCA untuk pengurangan dimensi fitur. Berbeda dari kedua penelitian sebelumnya, [12] melakukan proses *retraining* dari model CNN yang sudah ada ke dalam bidang *image retrieval* dengan mengubah parameter yang ada.

Untuk mendapatkan hasil klasifikasi yang lebih baik, [13] mengimplementasikan proses modulasi filter Gabor dengan filter yang ada pada lapisan konvolusi CNN. Modulasi tersebut bertujuan untuk meningkatkan akurasi pada CNN. Selama proses pelatihan filter gabor akan mengalami proses update seperti filter konvolusi pada umumnya. Sementara itu, [6] menggantikan filter konvolusi pada CNN dengan fungsi Gabor dibanding dengan fungsi acak. Penelitian tersebut menguji hasil pelatihan antara filter gabor yang dilatih dengan yang tidak dilatih. Hasilnya, penggunaan filter gabor yang dilatih mendapatkan hasil akurasi yang lebih baik.

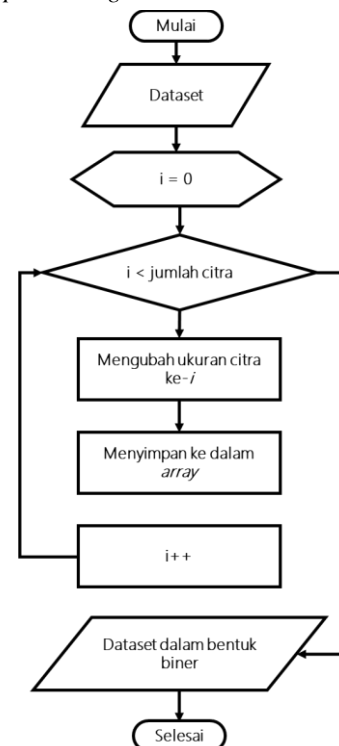
Berdasarkan pernyataan-pernyataan tersebut, maka penelitian ini akan berusaha untuk memanfaatkan dan mengimplementasikan metode CNN pada bidang *image retrieval*. Penelitian akan berfokus pada performa dari CBIR dengan fitur yang didapat dari pelatihan CNN. Penggunaan kernel berupa filter Gabor juga akan dimanfaatkan pada

lapisan konvolusi CNN. Tujuan digunakannya filter ini adalah untuk mendapatkan tekstur dari citra. Penggunaan filter ini juga dianggap mampu menyemai persepsi visual manusia dalam melakukan ekstraksi fitur.

II. METODOLOGI PENELITIAN

Pada penelitian ini, pendekatan yang dilakukan dalam bidang CBIR adalah melalui penggunaan metode CNN. CNN akan dilatih terlebih dahulu demi mendapatkan filter yang optimal dan berfungsi untuk mendapatkan fitur dengan performa yang baik pada proses pengembalian citra.

A. Data Preprocessing



Gbr. 1 Alur Data Preprocessing

Gbr. 1 menunjukkan alur dari *data preprocessing*. *Data preprocessing* akan dilakukan dengan mengubah citra berformat JPEG menjadi berkas biner. Pemrosesan ini bertujuan agar program tidak perlu membaca satu per satu citra, sehingga membutuhkan waktu yang lebih lama. Pada penelitian ini, kami menggunakan dataset GHIM10k [5] untuk menguji model CBIR yang akan dirancang. Dataset ini berisi 10000 citra dalam 20 kategori, dimana masing-masing kelas berisi 500 citra dengan ukuran 300 x 400 piksel dan 400 x 300 piksel. Sebelum melakukan pelatihan, maka ukuran citra akan dinormalisasi menjadi 227 x 227 piksel. Jumlah data citra yang digunakan pada proses pelatihan adalah 7500 citra. Gbr. 2 menunjukkan contoh beberapa citra yang ada pada dataset GHIM10k. Sesuai pada gambar tersebut, dataset ini memiliki

citra dengan tekstur yang beragam. Pada penelitian ini, pengambilan data latih dan data uji dilakukan secara acak.



Gbr. 2 Contoh data citra pada dataset GHIM10k

B. CNN

CNN merupakan jenis *neural network* yang bekerja pada data dengan dimensi tinggi [14]. Sama halnya dengan *neural network*, CNN mengandung neuron yang saling terhubung pada *visual field* [15]. Umumnya, CNN terdiri atas beberapa lapisan, yakni lapisan konvolusi, *pooling*, dan *fully connected* [8]. Lapisan konvolusi dan *pooling* merupakan lapisan yang bertujuan untuk mempelajari fitur pada citra, sedangkan *fully connected layer* akan melakukan klasifikasi citra.

1. Lapisan Konvolusi

Lapisan yang paling dikenal pada CNN adalah lapisan konvolusi. Lapisan ini akan melakukan konvolusi terhadap data masukan atau dari lapisan sebelumnya dengan menggeser sebuah filter untuk menghasilkan *feature map*. Namun pada kenyataannya, satu *feature map* saja tidak cukup untuk mengetahui karakter dari konten sebuah citra. Karena itu, penggunaan banyak filter dilakukan pada CNN [15]. Persamaan (1) menunjukkan operasi matematis dari konvolusi.

$$x = \varphi(W * Y_{(ij)} + b) \quad (1)$$

Dimana :

- Y = data input
- $\varphi(\cdot)$ = fungsi aktivasi
- b = bias
- W = *weight*/bobot

Pada proses pelatihan, bobot pada lapisan konvolusi akan mengalami proses *update* untuk memperbaiki hasil klasifikasi citra. proses *update* ini akan dilakukan terhadap semua bobot pada semua lapisan konvolusi. Pemilihan fungsi aktivasi juga menjadi kunci dari keberhasilan pelatihan CNN.

2. Lapisan *Pooling*

Lapisan konvolusi akan menghasilkan *feature map* yang masih cukup besar. Hal ini bisa berpengaruh pada kecepatan komputasi saat pelatihan. Lapisan *pooling* adalah salah satu cara untuk mengatasi permasalahan tersebut. Lapisan ini akan meringkas data masukan atau

feature map menjadi data yang lebih kecil. Umumnya, terdapat dua jenis *pooling*, yakni *max pooling* dan *average pooling*. *Max pooling* akan meringkas data dengan mencari nilai terbesar pada setiap kali penggeseran *window pooling*. Pada *average pooling*, akan dihitung nilai rata-rata tiap piksel pada setiap kali penggeseran. Ilustrasi dari *pooling* ditunjukkan pada Gbr. 3.

1	3	0	12
-2	0	3	-3
2	-3	3	-1
4	3	5	-5

3	12
4	5

Fitur

Hasil pooling

Gbr. 3 Proses *pooling*

3. Lapisan *Fully Connected*

Setelah mendapatkan fitur dari lapisan-lapisan sebelumnya, maka data akan masuk ke dalam tahap klasifikasi. Tahap ini akan dilakukan oleh lapisan *fully connected*. Lapisan ini sebenarnya adalah *multi layer perceptron* yang memiliki lapisan input, lapisan tersembunyi, dan lapisan output. Setiap lapisan memiliki neuron yang terhubung dengan lapisan berikutnya. Neuron-neuron ini akan memiliki *weight* yang akan membantu pada proses pemetaan data ke dalam kelasnya masing-masing. Pada penelitian ini, lapisan *fully connected* tidak akan digunakan pada proses *retrieval*.

4. Filter Gabor

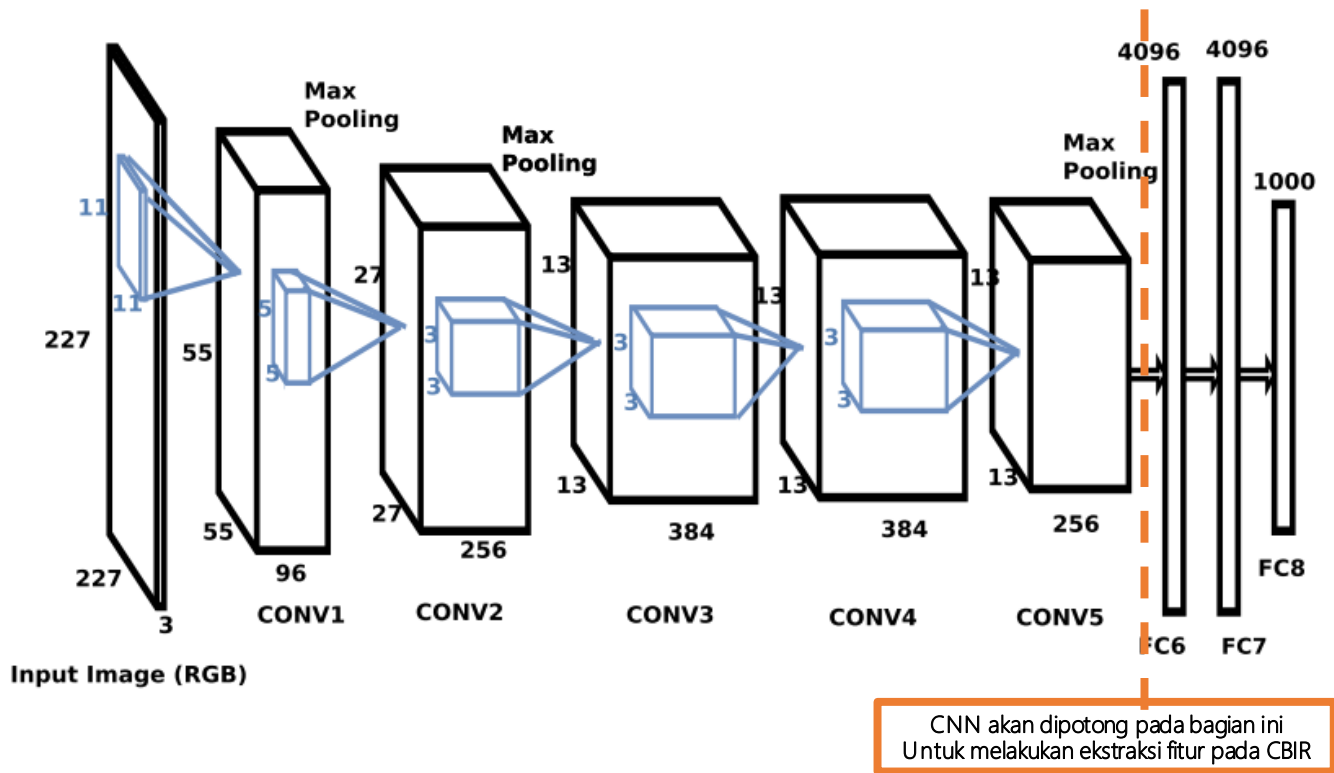
Filter Gabor merupakan jenis filter yang banyak digunakan untuk mengekstrak fitur dari sebuah citra, terutama fitur dalam bentuk tekstur [16]. Filter ini banyak digunakan dalam aplikasi analisis pola seperti, pengenalan karakter, pengenalan objek, dan tracking. Representasi frekuensi dan orientasi filter Gabor mirip dengan yang ada pada sistem visual manusia, dan dianggap sangat sesuai untuk representasi tekstur [17]. Pada penelitian ini, filter gabor akan diletakkan pada CNN. Filter gabor akan menggantikan filter pada *convolution layer* CNN yang biasanya menggunakan fungsi acak. Fungsi Gabor didefinisikan pada persamaan (2) dengan x' seperti pada persamaan (3) dan y' seperti pada persamaan (4).

$$g(x, y) = \exp\left(-\frac{x'^2 + y'^2}{2\sigma^2}\right) \cos\left(\frac{2\pi x'}{\lambda} + \psi\right) \quad (2)$$

dengan :

$$x' = x \cos \theta + y \sin \theta \quad (3)$$

$$y' = -x \sin \theta + y \cos \theta \quad (4)$$

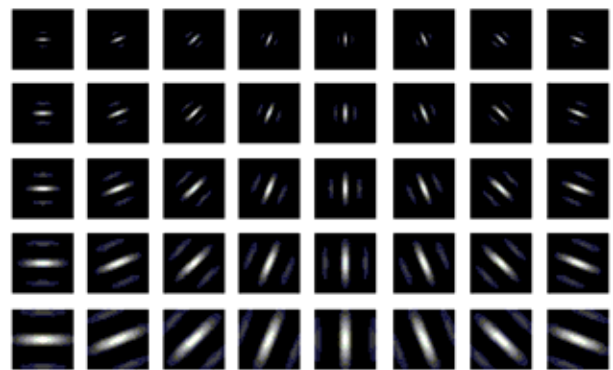


Gbr. 4 Arsitektur AlexNet [12]

dimana,

- (x, y) = lokasi piksel pada domain spasial
- λ = panjang gelombang sinus
- θ = orientasi (sudut) dari filter gabor
- ψ = fase *offset* dari fungsi sinus
- σ = standar deviasi dari fungsi Gaussian
- γ = aspek rasio dari fungsi Gabor

Sesuai pada Gbr. 5, filter Gabor memiliki orientasi dan skala yang berbeda. Parameter-parameter pada fungsi Gabor akan memengaruhi bentuk dari filter yang dihasilkan. Terdapat beberapa parameter pada fungsi Gabor yakni *theta*, *lamda*, *sigma*, *psi*, dan *gamma*. Parameter *theta* akan mengatur sudut atau derajat fungsi Gabor. Panjang pendeknya *bar* dari filter Gabor dipengaruhi oleh parameter *gamma*. Parameter *sigma* akan berpengaruh pada jumlah *bar* atau *strip* pada filter Gabor. Parameter *gamma* akan memengaruhi lebar tipisnya *bar* pada filter Gabor. Sementara itu, *psi* mengatur pergeseran pada fungsi sinus. Pada penelitian ini, ukuran dari filter Gabor akan disesuaikan dengan panjang dan lebar dari setiap filter pada lapisan konvolusi yang sesuai dengan arsitektur AlexNet.



Gbr. 5 Contoh filter Gabor [18]

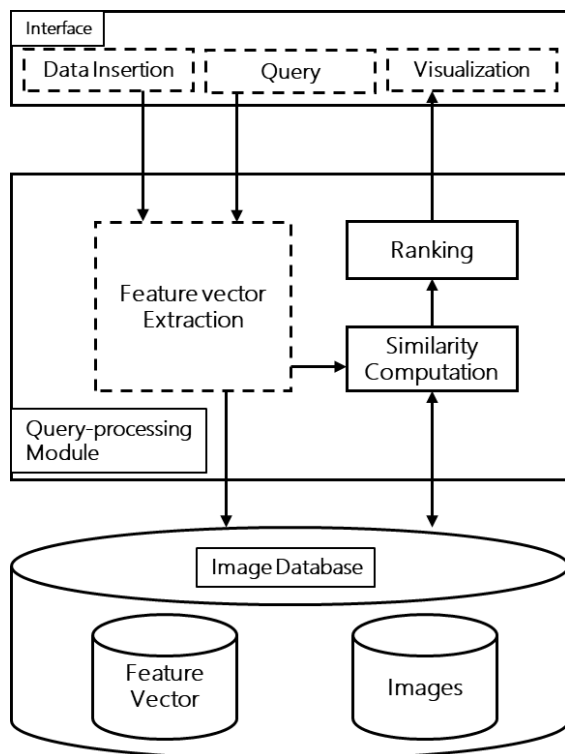
Pada penelitian ini, kami akan menggunakan arsitektur AlexNet yang sesuai pada Gbr. 4. Arsitektur ini mengandung 5 lapisan konvolusi, 3 lapisan *pooling*, dan 2 lapisan tersembunyi pada *fully connected*. Berbeda dengan Gbr. 4, jumlah neuron pada lapisan keluaran pada penelitian ini sesuai dengan jumlah kelas pada dataset GHIM10k, yakni 20 neuron saja. Dalam proses pelatihan CNN, kami juga akan mengimplementasikan penggunaan *dropout* dan *Batch Normalization*. Penggunaan *dropout* ini diharapkan mampu

mengurangi efek *overfitting* seperti pada [19]. Sementara itu, penggunaan *Batch Normalization* pada [10] dianggap efektif dalam mengurangi efek dari *internal covariate shift* saat proses pelatihan CNN.

C. Pengembalian Citra

Tahapan terakhir pada penelitian ini adalah dengan menguji hasil pengembalian citra. Pengujian dapat dilakukan setelah model *feature extractor* sudah dibuat dan dianggap telah memadai. Dalam kasus ini, artinya model CNN yang telah dirancang telah dilatih sehingga mampu mendapatkan fitur yang optimal dari citra yang masuk. Proses pengembalian citra ini dilakukan dalam beberapa tahap seperti pada Gbr. 6.

Kueri yang masuk dalam sistem akan diolah dengan model CNN yang telah terlatih sebelumnya. Model CNN tersebut akan menghasilkan fitur yang merupakan keluaran pada lapisan konvolusi, atau lebih tepatnya pada *flatten layer*. Jumlah fitur yang dihasilkan sangat dipengaruhi oleh bentuk dari arsitektur CNN yang dilatih. Hasil dari CBIR juga bergantung dari jumlah fitur yang digunakan. Semakin kecil jumlah fitur, maka komputasi yang dilakukan akan semakin cepat. Namun, semakin kecil jumlah fitur tidak menunjukkan semakin bagus pula hasil dari CBIR.



Gbr. 2 Arsitektur CBIR secara umum [11]

1. Penghitungan jarak atau kemiripan

Fitur yang telah didapat dari citra kueri akan dihitung kemiripannya dengan setiap fitur pada basis data atau

daftar citra koleksi. Fitur pada basis data merupakan fitur yang didapat dari data latih CNN. Selain pada jenis ekstraksi fitur, hasil dari CBIR juga sangat bergantung pada pemilihan pengukuran jarak. Untuk mengetahui pengukuran jarak yang efektif pada CBIR, maka penelitian ini akan membandingkan hasil pengembalian citra yang diuji dengan beberapa metode pengukuran jarak. *Euclidean* merupakan pengukuran jarak dengan menghitung akar kuadrat dari selisih fitur citra seperti yang ditunjukkan pada persamaan (5).

$$D(Q, C) = \sqrt{\sum_{i=1}^n (Q_i - C_i)^2} \quad (5)$$

Penggunaan *cosine distance* juga akan digunakan pada penelitian ini. Berbeda dengan *euclidean*, pengukuran jarak ini akan mencari besaran sudut dari dua vektor. *Cosine* didefinisikan pada persamaan (6).

$$D(Q, C) = \frac{\sum_{i=1}^l Q_i C_i}{\sqrt{\sum_{i=1}^l Q_i^2} \sqrt{\sum_{i=1}^l C_i^2}} \quad (6)$$

Selain kedua pengukuran jarak tersebut, penelitian ini juga menguji performa CBIR dengan pengukuran jarak *canberra distance*. *Canberra distance* didapatkan dengan membagi harga mutlak dari selisih fitur dua citra dengan penjumlahan fitur dua citra tersebut seperti yang didefinisikan pada persamaan (6). Semakin banyak jumlah fitur, maka semakin tinggi pula nilai jaraknya [20].

$$D(Q, C) = \sum_{i=0}^m \frac{|Q_i - C_i|}{|Q_i| + |C_i|} \quad (7)$$

2. Pengurutan atau perangkingan

Untuk mengetahui citra yang paling mirip dengan citra kueri, maka akan dilakukan pengurutan jarak. Pengurutan ini dilakukan dari jarak yang paling kecil hingga terbesar. Semakin kecil jarak sebuah citra, maka semakin mirip pula kedua citra tersebut. Namun, kemiripan ini adalah persepsi dari sebuah sistem. Kemiripan secara visual oleh manusia bisa berbeda dengan sistem. Karena itu, penelitian ini berusaha untuk merancang model CBIR yang dianggap mirip dengan persepsi visual manusia.

3. Evaluasi

Proses evaluasi bertujuan untuk mengetahui performa dari rancangan model CBIR yang dibuat. Umumnya, evaluasi CBIR dilakukan dengan menghitung nilai presisi. Jika *RC* adalah jumlah citra yang dikembalikan dan relevan terhadap citra kueri dan *A* adalah jumlah citra yang dikembalikan maka, sesuai pada persamaan (8) presisi dapat dihitung dengan membagi *RC* dengan *A*.

$$Precision = \frac{RC}{A} \quad (8)$$

Karena jumlah data uji yang digunakan beragam, maka kami menggunakan pendekatan nilai mAP (*mean average precision*) untuk mengetahui sejauh mana model CBIR kami dapat melakukan pengembalian citra yang sesuai dengan harapan. Nilai mAP ini didapat dengan menghitung rata-rata nilai presisi dari setiap data uji atau citra kueri. Semakin tinggi nilai mAP, maka semakin baik pula CBIR yang dirancang. Nilai mAP dapat dihitung dengan menggunakan persamaan (9)

$$mAP = \frac{1}{K} \sum_{i=1}^K \frac{1}{Q_i} \sum_{n=1}^N \frac{R_i^n}{n} t_n^i \quad (9)$$

Dimana K adalah jumlah kueri yang diuji, Q_i adalah kueri ke- i , N adalah jumlah total citra yang dikembalikan, R_i^n adalah jumlah citra yang relevan terhadap n citra pertama yang dikembalikan, t_n^i adalah fungsi indikator, dimana $t_n^i = 1$ jika citra ke- n relevan terhadap citra kueri, dan $t_n^i = 0$ jika sebaliknya.

$$Akurasi = \frac{\text{Jumlah data benar}}{\text{Jumlah keseturuhan data}} * 100\% \quad (10)$$

Untuk menguji model CNN, penelitian ini menggunakan akurasi seperti pada persamaan (10) untuk mengetahui performa dari klasifikasi citra. semakin tinggi nilai akurasi, maka semakin baik pula model klasifikasi citra yang dirancang. Model CBIR yang dibuat bergantung pada akurasi yang dihasilkan oleh CNN, karena filter yang terlatih yang menghasilkan akurasi yang tinggi dapat dengan baik untuk mendapatkan fitur citra pada proses ekstraksi fitur.

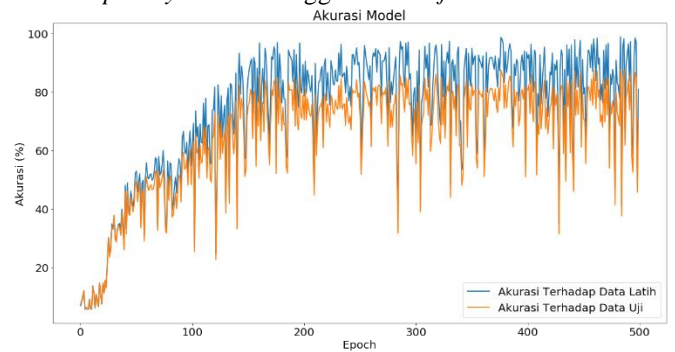
III. HASIL DAN PEMBAHASAN

Penelitian dilakukan pada platform *kaggle.com* dengan bahasa pemrograman Python. Platform tersebut menggunakan GPU NVIDIA Tesla K80 dengan CPU Intel Xeon 16 Cores @ 2.3 GHz.

A. Pelatihan CNN

Pelatihan CNN dilakukan dengan 500 *epoch*. Metode *pooling* yang dilakukan adalah *average pooling*. Ukuran *pooling* pada lapisan konvolusi terakhir memiliki ukuran yang berbeda, yakni 13 x 13 atau sesuai dengan ukuran keluaran dari lapisan konvolusi kelima. Pengubahan ukuran ini dilakukan untuk mengurangi fitur yang akan digunakan pada CBIR dibanding dengan menggunakan *dimensionality reduction*. *RMSprop* digunakan sebagai *optimizer*, sedangkan fungsi *loss* yang dipakai adalah *categorical cross-*

entropy. Fungsi aktivasi yang digunakan pada setiap lapisan berbeda. *ReLU* akan diletakkan pada lapisan konvolusi, *sigmoid* untuk lapisan tersembunyi pada *fully connected layer*, dan *output layer* akan menggunakan *softmax*.



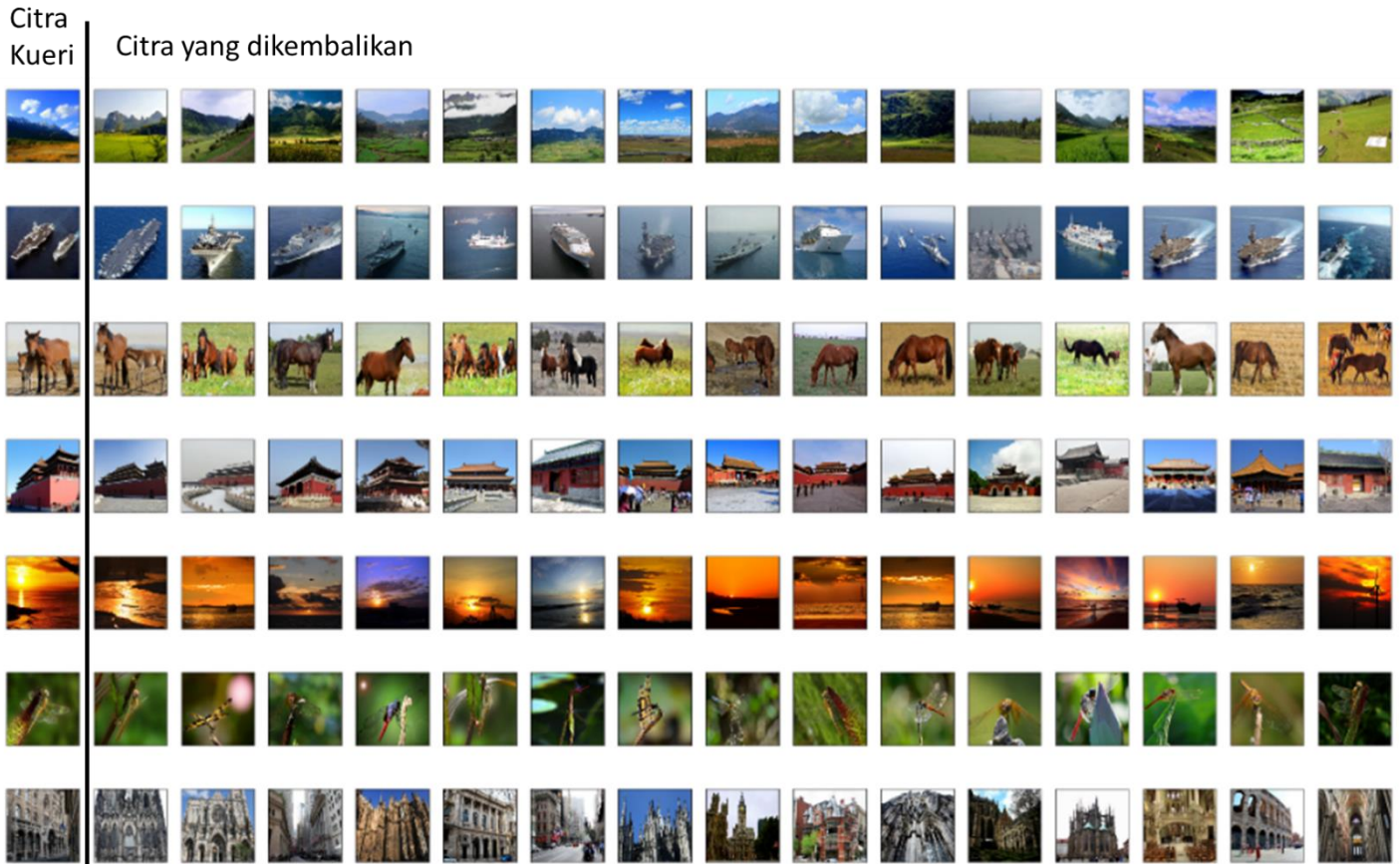
Gbr. 7 Akurasi tiap epoch pada pelatihan CNN

Gbr. 7 menunjukkan bahwa pergerakan akurasi tidak stabil saat pelatihan. Hal ini bisa terjadi karena bentuk data yang variatif. Pelatihan CNN yang dilakukan mendapat akurasi sebesar 88,12% terhadap data uji dan 99,093% terhadap data latih. Akurasi tersebut didapatkan dengan penggunaan nilai *dropout* sebesar 0,6 pada setiap lapisan setelah *pooling*, dan 0,2 pada lapisan tersembunyi di *fully connected layer*. Sementara itu, *Batch Normalization* diletakkan pada setiap lapisan setelah lapisan konvolusi. Setelah pelatihan CNN selesai, model akan disimpan untuk digunakan kembali pada sistem CBIR.

Pada sistem CBIR, CNN tidak akan digunakan sepenuhnya dalam melakukan ekstraksi fitur. Pada penelitian ini, kami memotong CNN pada bagian *flatten layer*, sehingga fitur yang didapat adalah hasil dari *layer* tersebut. Penggunaan *pooling* dengan ukuran maksimum (sesuai dengan tinggi dan lebar *layer* konvolusi terakhir), maka jumlah fitur yang didapat adalah 256 fitur saja. Sebelum masuk ke pengujian *image retrieval*, proses ekstraksi fitur akan dilakukan terhadap seluruh citra yang digunakan sebagai data latih. Citra-citra inilah yang akan dikembalikan pada proses CBIR nantinya. Fitur-fitur ini akan disimpan agar tidak perlu melakukan ekstraksi fitur kembali setiap kali melakukan pencarian.

B. Pengembalian Citra

Pengembalian citra akan dilakukan setelah CNN terlatih dengan baik. CNN yang telah terlatih memiliki filter pada konvolusi yang sudah mampu mengenali pola citra dari dataset. Fitur yang digunakan pada proses pengembalian citra adalah hasil keluaran dari *flatten layer* pada CNN. Fitur pada *flatten layer* dipilih untuk mengurangi biaya komputasi yang dilakukan pada *fully connected*. Selain itu, jumlah fitur yang dihasilkan lebih sedikit, sehingga mampu mempercepat proses penghitungan jarak. Jumlah fitur yang digunakan pada penelitian ini adalah 256. Citra yang akan dikembalikan merupakan citra yang digunakan pada data latih. Tahap



Gbr. 8 Contoh hasil pengembalian citra dari beberapa citra kueri

klasifikasi pada CNN tidak akan digunakan saat proses pengembalian. Karena tidak ada tahap klasifikasi, maka citra kueri akan dihitung kemiripannya terhadap keseluruhan citra pada data latih. Proses pengembalian citra tidak melibatkan tahap klasifikasi agar keseluruhan citra pada dataset dapat melalui proses penghitungan jarak. Selain itu, jika citra kueri mengalami kesalahan klasifikasi, maka citra yang dikembalikan agar seluruhnya tidak relevan, sehingga berimbas pada menurunnya nilai mAP.

Pada pengujian pertama, penelitian ini menghitung nilai mAP dengan pengukuran jarak *euclidean distance*. Nilai mAP terhadap data uji mendapatkan hasil yang lebih besar dibanding akurasi pada CNN. Hal tersebut seperti yang ditunjukkan pada Tabel I. Tabel ini menunjukkan bahwa semakin banyak citra yang dikembalikan, maka nilai mAP juga akan semakin kecil.

TABEL I
 NILAI MAP TERHADAP DATA UJI DENGAN MENGGUNAKAN EUCLIDEAN DISTANCE

Jumlah Citra yang Dikembalikan	mAP
20	0,886

40	0,880
60	0,875
80	0,870
100	0,866

Hasil yang berbeda ditunjukkan pada Tabel II. Penggunaan *cosine distance* mendapatkan nilai mAP yang lebih tinggi dibanding dengan *euclidean distance*. Sama halnya dengan tabel I, nilai mAP akan semakin kecil seiring dengan bertambahnya jumlah citra yang diterima. Hal tersebut bisa terjadi karena semakin banyaknya citra yang dikembalikan, maka semakin banyak pula kesempatan bagi citra yang tidak relevan untuk dikembalikan ke pengguna.

Penggunaan *cosine distance* terhadap data uji mendapat nilai mAP yang lebih tinggi pula daripada pada penggunaan *euclidean distance* seperti yang ditunjukkan pada Tabel II. Bahkan, selisih yang didapatkan bisa dikatakan cukup besar. Pada jumlah citra yang dikembalikan sebanyak 20 buah saja, selisihnya mencapai 0,13 dimana pada *cosine distance* mendapatkan nilai mAP sebesar 0,893. Nilai mAP ini bahkan lebih tinggi dibanding dengan nilai akurasi pada CNN. Bahkan pada jumlah citra yang dikembalikan sebanyak 100 buah, nilai mAP masih

bertahan di angka 0,882 dimana pada penggunaan *euclidean distance* mAP hanya mencapai 0,866 saja.

TABEL II
NILAI MAP TERHADAP DATA UJI DENGAN MENGGUNAKAN COSINE DISTANCE

Jumlah Citra yang Dikembalikan	mAP
20	0,893
40	0,888
60	0,886
80	0,883
100	0,882

Tabel III menunjukkan hasil pengembalian citra dengan menggunakan *canberra distance* dengan data uji sebagai citra kueri. Dari penggunaan pengukuran jarak ini, didapatkan peningkatan nilai mAP jika dibandingkan dengan penggunaan kedua pengukuran jarak sebelumnya. Pada proses pengembalian dengan jumlah citra yang dikembalikan sebanyak 100 buah, mAP dengan *canberra distance* masih lebih tinggi. Penggunaan *canberra distance* juga memiliki performa yang lebih unggul. Performa pada pengukuran jarak ini bahkan melampaui nilai akurasi pada model CNN. Pada proses pengembalian dengan jumlah citra yang dikembalikan sebanyak 20 buah, mAP dengan *canberra distance* mendapat mAP sebesar 0,895. Dengan jumlah citra yang dikembalikan sebanyak 100 buah saja, nilai mAP masih melampaui nilai akurasi, yakni sebesar 0,885.

TABEL III
NILAI MAP TERHADAP DATA UJI DENGAN MENGGUNAKAN CANBERRA DISTANCE

Jumlah Citra yang Dikembalikan	mAP
20	0,895
40	0,891
60	0,888
80	0,886
100	0,885

Pada penggunaan *cosine* dengan jumlah citra yang dikembalikan sebanyak 20 buah, waktu rata-rata waktu untuk melakukan pengembalian citra adalah 0,326 detik. Sedangkan pada *euclidean*, rata-rata waktu untuk melakukan pengembalian citra lebih singkat, yakni hanya 0,119 detik dan penggunaan *canberra distance* memakan waktu 0,318 detik. Dalam hal penghitungan jarak, *euclidean distance* membutuhkan waktu yang lebih singkat dibanding kedua pengukuran jarak lainnya.

Gbr. 8 menunjukkan contoh beberapa hasil pengembalian citra pada sistem CBIR yang dibuat. Hasilnya, sistem mampu mengembalikan beberapa citra yang memiliki gambaran visual yang mirip dengan citra kueri. Walaupun proses penghitungan jarak dilakukannya terhadap seluruh citra pada dataset, namun citra relevan dengan citra kueri mampu dikembalikan dengan baik.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan oleh peneliti diperoleh kesimpulan bahwa penggunaan CNN dengan filter gabor sebagai *feature extractor* mampu mendapatkan hasil yang cukup baik dalam bidang CBIR dengan dataset GHIM10k. Hal ini didasarkan pada nilai akurasi CNN yang mencapai 88,12% terhadap data uji dan nilai mAP sebesar 0,895 terhadap data uji pada jumlah citra yang dikembalikan sebanyak 20 buah pada penggunaan pengukuran jarak *canberra distance*. Peneliti juga mendapati bahwa penggunaan *canberra distance* mendapatkan hasil yang lebih baik dibanding dengan menggunakan *euclidean distance* dan *cosine distance*. Waktu pengembalian citra yang dibutuhkan adalah 0,318 detik.

CBIR merupakan bidang penelitian yang masih menjadi hal yang menarik untuk dikembangkan. Peneliti berharap penelitian mengenai CBIR dengan CNN dapat dikembangkan kedepannya dengan mencoba berbagai model CNN yang ada atau dengan metode *hybrid* dari CNN untuk mendapatkan model dengan performa yang lebih baik. Peneliti juga berharap agar dataset yang digunakan lebih beragam.

REFERENSI

- [1] Y. Mistry, D. Ingole dan M. Ingole, "Content based image retrieval using hybrid features and various distance metrics," *Electrical Systems and Information Technology*, pp. 1-15, 2016.
- [2] S. Unar, W. Xingyuan dan Z. Chuan, "Visual and Textual Information Fusion Using Kernel Method for Content Based Image Retrieval," *Information Fusion*, 2018.
- [3] X.-Y. Ou, H.-F. Ling, L.-Y. Yan dan M.-L. Liu, "Convolutional Neural Codes for Image Retrieval," dalam *Signal and Information Processing Association Annual Summit and Conference*, Siem Reap, 2014.
- [4] M. J. Darugar dan L. C. Kiong, "Gender Estimation based on Supervised HOG, Action Units and Unsupervised CNN Feature Extraction," *Artificial Intelligence and Robotics*, pp. 23-27, 2017.
- [5] G.-H. Liu, J.-Y. Yang dan Z. Li, "Content-based image retrieval using computational visual attention model," *Pattern Recognition*, 2015.
- [6] S. S. Sarwar, P. Panda dan K. Roy, "Gabor Filter Assisted Energy Efficient Fast Learning Convolutional Neural Networks," Taipei, Taiwan, 2017.
- [7] K. T. Ahmed, Shahida dan M. A. Iqbal, "Content Based Image Retrieval using Image Features Information Fusion," *Information Fusion*, 2018.
- [8] T. Guo, J. Dong, H. Li dan Y. Gao, "Simple Convolutional Neural Network on Image Classification," dalam *International Conference on Big Data Analysis*, Beijing, 2017.
- [9] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever dan R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research* 15, pp. 1929-1958, 2014.

- [10] S. Ioffe dan C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," 2015.
- [11] A. Babenkao, A. Slesarev, A. Chigorin dan V. Lempitsky, "Neural Codes for Image Retrieval," dalam *ECCV*, Zurich, 2014.
- [12] M. Tzelepi dan A. Tefas, "Deep Convolutional Learning for Content Based Image Retrieval," *Neurocomputing*, vol. 275, pp. 2667-2478, 2018.
- [13] C. C. B. Z. J. H. d. J. L. S. Luan, "Gabor Convolutional Networks," *IEEE Transactions on Image Processing*, vol. 27, pp. 4357-4366, 29 Januari 2018.
- [14] S. Khan, H. Rahmani, S. A. A. Shah dan M. Bennamoun, *A Guide to Convolutional Neural Networks for Computer Vision*, Morgan & Claypool Publishers, 2018.
- [15] O. Mohamed, E. A. Khalid, O. Mohamed dan A. Brahim, "Content-Based Image Retrieval Using Convolutional Neural Networks," pp. 463-476, 2019.
- [16] V. Tyagi, *Content-Based Image Retrieval: Ideas, Influences, and Current Trends*, London: Springer, 2018.
- [17] C. Youness, E. A. Khalid, O. Mohammed dan A. Brahim, "Content-Based Image Retrieval Using Gabor Filters and 2-D ESPRIT Method," 2017.
- [18] A. Vinay, V. S. Shekhar, K. Murthy dan S. Natarajan, "Performance Study of LDA and KFA for Gabor Based Face Recognition System," 2015.
- [19] H. Wu dan X. Gu, "Towards dropout training for convolutional neural networks," *Neural Networks*, vol. 71, pp. 1 - 10, 2015.
- [20] M. C. P. dan P. K., "A Survey on Feature Selection Stability Measures," *International Journal of Computer and Information Technology*, vol. 05, no. 1, pp. 98 - 103, 2016.