

# FAST IMAGE RETRIEVAL BERBASIS LOCALITY SENSITIVE HASHING DAN CONVOLUTIONAL NEURAL NETWORK

Silvester Tena<sup>1</sup>, Bernadectus Yudi Dwiandiyanta<sup>2</sup>, Wenefrida Tulit Ina<sup>3</sup>

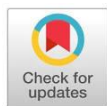
<sup>1,3</sup>Program Studi Teknik Elektro, Fakultas Sains dan Teknik, Universitas Nusa Cendana, Kupang, Indonesia

<sup>2</sup>Program Studi Informatika, Fakultas Teknologi Industri, Universitas Atma Jaya Yogyakarta, Yogyakarta, Indonesia

Email: siltena@staf.undana.ac.id, yudi.dwiandiyanta@uajy.ac.id, wenefrida\_ina@staf.undana.ac.id

## Info Artikel

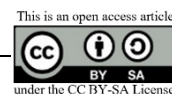
Article History:  
Received Feb 02, 2024  
Revised Apr 13, 2024  
Accepted Apr 30, 2024



## ABSTRACT

Image retrieval systems with a fast search process are still challenging for researchers. Fast search methods are one of the most important parts of image retrieval. One of the techniques used is reducing feature dimensions using the Locality Sensitivity Hashing (LSH) method. Apart from that, feature types and image extraction methods are selected. Image feature extraction uses the Convolutional Neural Network (CNN) method in this research. Measuring similarity using the Hamming Distance (HD) and Euclidean Distance (ED) methods. The datasets used are TenunIkatNet and Batik300. The LSH method forms a hash table as a bucket to group similar images based on probability and in the form of binary code. The testing process begins with Hamming Distance to find buckets that match the binary code of the test image. The research results show that the LSH+HD+ED method provides faster search results than ED. The image retrieval time for the LSH+HD+ED and ED methods is 0.252 seconds and 4.5 seconds, respectively, for the TenunIkatNet dataset. Meanwhile, the Batik300 dataset is 0.03 seconds and 0.9 seconds. Using the LSH method is very effective for large datasets. Retrieval accuracy using the LSH+HD+ED method was 99.705% and 84% for the TenunIkatNet and Batik300 datasets, respectively. Meanwhile, the ED method produces 94.17% and 82% retrieval accuracy, respectively.

**Keywords:** image retrieval, CNN, LSH, ED, HD



## ABSTRAK

Sistem temu kembali citra (image retrieval) dengan proses pencarian yang cepat masih menjadi tantangan bagi peneliti. Metode pencarian yang cepat menjadi salah satu bagian terpenting dalam retrieval citra. Salah satu teknik yang digunakan adalah mengurangi dimensi fitur dengan metode Locality Sensitivity Hashing (LSH). Selain itu adalah pemilihan jenis fitur dan metode ekstraksi citra. Dalam penelitian ini ekstraksi fitur citra menggunakan metode Convolutional Neural Network (CNN). Pada proses pengukuran kemiripan menggunakan metode Hamming Distance (HD) dan Euclidean Distance (ED). Dataset yang digunakan yakni TenunIkatNet dan Batik300. Metode LSH membentuk tabel hash berupa bucket untuk mengelompokkan citra yang mirip berdasarkan probabilitas dan dalam bentuk kode biner. Proses pengujian diawali dengan Hamming Distance untuk mencari bucket yang sesuai dengan kode biner citra uji. Hasil penelitian menunjukkan bahwa metode LSH+HD+ED memberikan hasil pencarian lebih cepat dibandingkan dengan ED. Waktu retrieval citra metode LSH+HD+ED dan ED berturut-turut sebesar 0,252 detik dan 4,5 detik untuk dataset TenunIkatNet. Sedangkan dataset Batik300 yakni 0,03 detik dan 0,9 detik. Penggunaan metode LSH sangat efektif untuk dataset yang jumlahnya besar. Akurasi retrieval menggunakan metode LSH+HD+ED berturut-turut sebesar 99,705% dan 84% untuk dataset TenunIkatNet dan Batik300. Sedangkan metode ED menghasilkan akurasi retrieval berturut-turut yakni 94,17% dan 82%.

**Keywords:** image retrieval, CNN, LSH, ED, HD

## Penulis Korespondensi:

Silvester Tena,  
Program Studi Teknik Elektro, Fakultas Sains dan Teknik,  
Universitas Nusa Cendana,  
Jl. Adisucipto, Penfui, Kupang  
Email: siltena@staf.undana.ac.id



## 1. PENDAHULUAN

Perkembangan informasi era *big data* dengan jumlah gambar digital di internet berkembang secara eksponensial. Bagaimana cara mencari gambar-gambar yang diinginkan dari banyak basis data (*database*) gambar telah menjadi masalah penting yang perlu segera dipecahkan [1], dan pencarian gambar berbasis konten (CBIR) dapat menyelesaikan masalah ini. CBIR adalah teknologi yang berbasis pada visi komputer dan digunakan untuk mencari gambar serupa dari basis data. Gambar-gambar diwakili oleh fitur-fitur berdimensi tinggi, yang menggunakan berbagai ukuran metrik jarak untuk menghitung kemiripan antar gambar. Kinerja sistem *retrieval* citra berbasis pada fitur bentuk, struktur, warna, atau fitur lain yang diekstrak dari gambar dan kualitas pencocokan. Proses terpenting dalam sistem CBIR yakni pemilihan metode ekstraksi fitur (*feature extraction*) dan pengukuran kemiripan (*similarity measure*) [2], [3]. Dalam aspek ekstraksi fitur, metode *Convolutional Neural Network* (CNN) dapat digunakan untuk mengekstraksi fitur. Pemilihan metode CNN karena keunggulannya dapat mengekstraksi semua fitur visual objek citra dan mendukung *dataset* besar. Penelitian terkait optimasi metode CNN untuk objek citra *fabric* menghasilkan akurasi *retrieval* meningkat [3], [4]. Penelitian sebelumnya menunjukkan bahwa metode CNN mengungguli metode ekstraksi fitur lainnya pada jenis citra antara lain kain (*fabric*) [5], batik [6] [7]. Dalam aspek pencarian citra dengan fitur gambar berdimensi tinggi hasil ekstraksi metode CNN akan meningkatkan akurasi *retrieval*, namun kompleksitas komputasionalnya juga akan meningkat. Penelitian sebelumnya membandingkan vektor fitur gambar *query* berdimensi tinggi dengan vektor fitur dari gambar dalam basis data satu per satu. Metode ini memakan ruang penyimpanan fitur dan waktu pencocokan.

Dalam rangka mengurangi beban komputasi dan mempercepat pencarian digunakan algoritma *hash* untuk mendapatkan kode *hash* biner yang jauh lebih rendah daripada dimensi aslinya. Kode *hash* yang dihasilkan dapat digunakan untuk membangun indeks yang efisien, yang tidak hanya mewujudkan reduksi dimensi data, tetapi juga meningkatkan efisiensi pencocokan fitur.

Pengukuran kemiripan (*similarity measure*) dalam sistem *image retrieval* juga bagian yang penting untuk menentukan dua gambar mirip atau sama. Pengukuran kemiripan akan mencocokkan dan membandingkan fitur-fitur antara citra *query* dan citra *database*. Dalam sistem *image retrieval* membutuhkan waktu yang banyak untuk

perhitungan kemiripan antar fitur. Penerapan kode *hashing* dan *Hamming Distance* diharapkan dapat mempercepat waktu *retrieval* [2], [3], [5]. Ekstraksi citra *query* dilakukan hanya sekali dan fitur yang diekstrak akan dihitung kemiripannya dengan semua fitur citra dalam *database* fitur [2], [3].

Beberapa penelitian sebelum hanya memperhatikan ekstraksi fitur dalam meningkatkan akurasi *retrieval* namun mengabaikan waktu *retrieval*. Penelitian Xiang *et. al* untuk *retrieval Wool Fabric Images Database* (WFID) menunjukkan bahwa waktu *retrieval* sebesar 15,3 detik [3]. Pada penelitian *retrieval* citra tenun ikat menghasilkan waktu *retrieval* sebesar 9,94 detik [8].

Penelitian ini bertujuan melakukan *retrieval* citra menggunakan metode CNN dan LSH untuk mempercepat waktu pencarian. Metode CNN berfungsi mengekstraksi fitur citra dan LSH untuk mereduksi ukuran fitur citra. Pada proses pengukuran kemiripan menggunakan metode *Hamming Distance* dan *Euclidean Distance*.

Kontribusi utama dalam penelitian ini adalah:

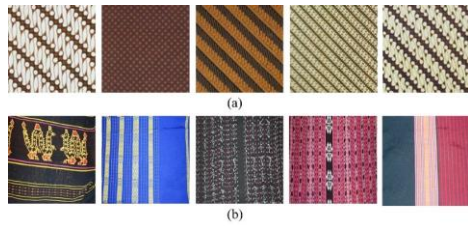
1. Proses augmentasi citra yakni *scaling*, *zooming*, *rotation*, dan *shear range* untuk meningkatkan jumlah data pelatihan. Proses Augmentasi bertujuan mengurangi *overfitting* pada proses pelatihan.
2. Vektor fitur citra hasil ekstraksi fitur CNN akan direduksi ukurannya menggunakan metode LSH dalam membangun indeks. Proses ini akan mempercepat pencarian citra uji dalam citra *database*.

Paper ini dibagi dalam beberapa bagian. Bagian 2 menjelaskan metode dan bahan penelitian yang digunakan secara rinci, termasuk langkah-langkah eksperimen yang dilakukan. Bagian 3 mencakup detail hasil eksperimen yang diperoleh dan pembahasan mendalam mengenai temuan yang ditemukan, ditambah dengan saran untuk penelitian lanjutan. Bagian terakhir dari paper ini merupakan kesimpulan, yang merangkum temuan utama dari penelitian ini.

## 2. METODE PENELITIAN

### 2.1 *Dataset* TenunIkatNet dan Batik

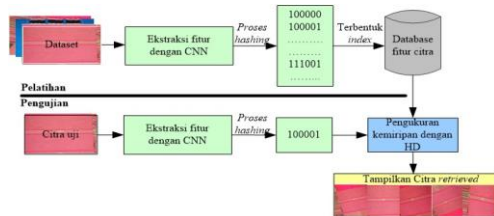
Dalam penelitian ini akan digunakan dua jenis *dataset* yakni TenunIkatNet [9] dan Batik [10]. *Dataset* TenunIkatNet terdiri dari 120 kelas. Tiap kelas sebanyak 40 citra dan jumlah totalnya adalah 4.800 citra. Ukuran citra kain tenun ikat adalah 256x256 piksel. *Dataset* batik terdiri dari 50 kelas. Tiap kelas memiliki 6 citra, sehingga jumlah totalnya 300 citra. Ukuran citra batik adalah 12x128. Gambar 1 menunjukkan contoh citra kain tenun ikat dan batik.



**Gambar 1.** Beberapa contoh citra, a) batik, b) kain tenun ikat.

**2.2 Kerangka Penelitian**

Kerangka penelitian *retrieval* citra kain tenun ikat dan batik dapat dilihat pada Gambar 2. Tahapan penelitian diawali dengan proses augmentasi citra untuk menambah *dataset* baik tenun ikat maupun batik. Augmentasi dilakukan untuk meningkatkan jumlah data pelatihan. *Dataset* yang dilengkapi dengan variasi hasil augmentasi akan dilakukan proses ekstraksi fitur pada proses pelatihan dan pengujian.

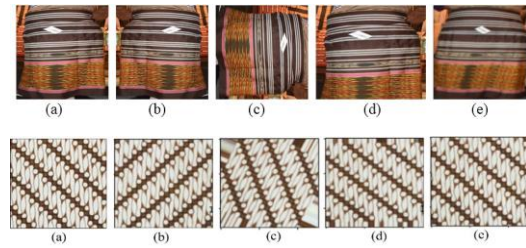


**Gambar 2.** Kerangka Penelitian

**2.3 Augmentasi citra**

Metode *deep learning* yakni CNN membutuhkan data pelatihan yang banyak untuk menghasilkan akurasi pembelajaran mesin yang tinggi. Beberapa jenis augmentasi standar yang digunakan adalah *zooming*, *random rotation*, *flipping*, dan *shear range* ditunjukkan pada Gambar 3. *Zooming* merupakan salah satu operasi geometri untuk efek memperbesar atau memperkecil ukuran citra masukan sesuai dengan variabel penskalaan citranya. *Random rotation* adalah proses rotasi citra

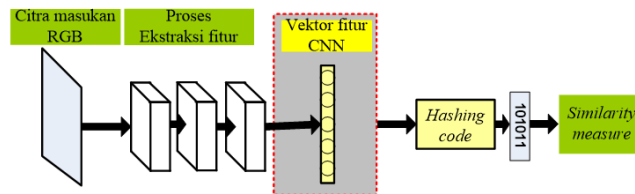
berdasarkan derajat tertentu. *Flipping* merupakan operasi geometri yakni pencerminan horisontal dan vertikal berturut-turut terhadap sumbu *Y* dan sumbu *X*. *Shear range* adalah menerapkan transformasi geser secara acak.



**Gambar 3.** Augmentasi citra kain tenun ikat dan batik, a) Citra asli, b) *Flipping*, c) *Rotating*, d) *Zooming*, dan e) *Shear range*.

**2.4 Ekstraksi Fitur dan Locality Sensitive Hashing**

Metode *Convolutional Neural Network* (CNN) telah menarik banyak perhatian dan penggunaannya semakin luas. Metode CNN berhasil dalam deteksi objek, klasifikasi gambar, *retrieval* citra, akuisisi pengetahuan, dan segmentasi semantik citra [7] [11]. Beberapa kelebihan CNN, yakni: 1) Secara otomatis mengekstraksi ciri penting dari setiap citra tanpa bantuan manusia, 2) Lebih efisien memori dan kompleksitas dibandingkan metode *neural network* lainnya, 3) *Parameter sharing* yang dapat membantu mengurangi jumlah parameter pada keseluruhan sistem dan membuat beban komputasinya berkurang, 4) Memiliki keunggulan *spatial features* yang mengacu pada susunan piksel dan hubungan antar piksel dalam sebuah citra. Vektor fitur hasil ekstraksi CNN akan dikonversi menjadi kode biner untuk mereduksi ukurannya. Metode *Locality Sensitive Hashing* digunakan dalam penelitian ini untuk mereduksi ukuran fitur sehingga mempercepat proses pengukuran kemiripan citra uji dan citra *database*.



**Gambar 4.** Proses ekstraksi fitur citra menggunakan metode CNN.

Metode CNN memiliki lapisan tersembunyi dan arsitektur yang lebih kompleks serta memiliki kemampuan pembelajaran fitur dan representasi fitur yang lebih canggih. Metode CNN merupakan algoritma pembelajaran mendalam (*deep learning*) yang dapat melatih kumpulan data besar dengan jutaan parameter dan mengambil bentuk gambar 2D

sebagai masukan, serta menggabungkannya dengan kernel untuk menghasilkan keluaran yang diinginkan. Metode Convolutional Neural Network (CNN) dirancang untuk secara otomatis dan adaptif mempelajari struktur hierarki spasial elemen dengan menggunakan propagasi mundur dari beberapa blok penyusun (seperti lapisan

konvolusional, lapisan aktivasi, lapisan pooling, menggunakan arsitektur yang ditunjukkan pada dan lapisan *fully connected*). Penelitian ini Tabel 1.

Tabel 1. Arsitektur CNN untuk ekstraksi fitur citra tenun ikat dan batik.

| Tipe layer              | Ukuran                                | Ukuran output                      |
|-------------------------|---------------------------------------|------------------------------------|
| input                   | (256,256,3)* (128,128, 3)**           |                                    |
| Conv2d + Relu           | 32 (3x3) filters, 1 stride, 0 padding | (254, 254, 32)*<br>(126,126, 32)** |
| Max Pooling             | 32 (2x2) filters, 2 stride, 0 padding | (127, 127, 32)*<br>(63, 63, 32)**  |
| Conv2d_1 + Relu         | 32 (3x3) filters, 1 stride, 0 padding | 125, 125, 32*<br>(61, 61, 32)**    |
| Max Pooling             | 32 (2x2) filters, 2 stride, 0 padding | (62, 62, 32)*<br>(30, 30, 32)**    |
| Conv2d_2+ Relu          | 32 (3x3) filters, 1 stride, 0 padding | (60, 60, 32)*<br>(28, 28, 32)**    |
| Max Pooling             | 32 (2x2) filters, 2 stride, 0 padding | (30, 30, 32)*<br>(14, 14, 32)**    |
| Flatten                 | (1,1, 28.800)<br>(1, 1, 6.272)        | 28.800*<br>6.272**                 |
| Dense                   | 512 neurons                           | 512                                |
| Dense_1 + Dropout (50%) | 256 neurons                           | 256                                |
| Softmax                 | 120 kelas*<br>50 kelas**              | 120*<br>50**                       |

Keterangan: \* = tenun, \*\* = batik.

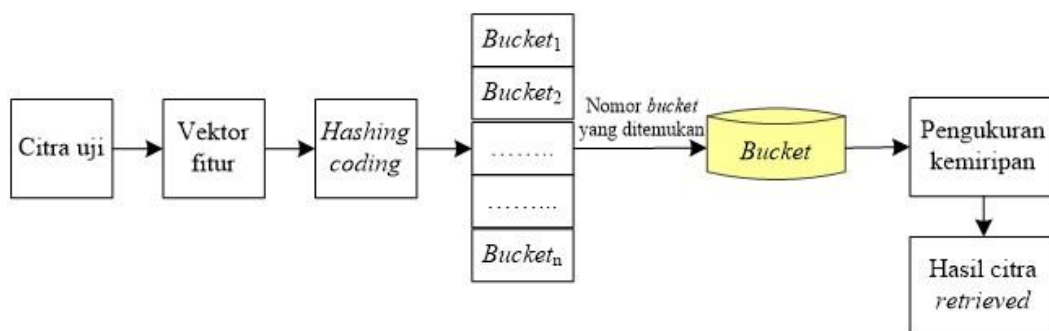
Dalam sistem *retrieval* citra waktu pengukuran kemiripan menjadi salah satu faktor yang harus diperhatikan. Vektor fitur hasil ekstraksi CNN akan dikonversi kedalam kode *hashing* dengan metode *Locality Sensitive Hashing* (LSH) baik untuk citra *database* maupun citra uji. Prinsip dasarnya adalah vektor fitur hasil ekstraksi model CNN akan dikelompokkan berdasarkan kemiripannya. Metode LSH dengan strategi *random projection* untuk membentuk vektor masukan. Pada pengujian membuat *indexing* terdiri dari 6 bit kode. Proses pembuatan *indexing* dengan LSH membutuhkan masukan berupa *hashing code*, dimensi vektor fitur, dan vektor fitur berupa probabilitas statistik hasil ekstraksi fitur dengan metode *feature extractor*. Pada pengujian ini diketahui bahwa *hashing code* ( $h$ ) = 6, dimensi fitur ( $dim$ )= 120, dan vektor fitur =  $p$ . Matriks *random projection* sebagai berikut:

$$Random\ projection\ (h_k) = (rand(h, dim)) \quad (1)$$

Untuk semua vektor fitur:

$$\begin{aligned} \text{Jika } h_k \bullet p < 0 &\rightarrow \text{nilai hash} = 0, \\ h_k \bullet p > 0 &\rightarrow \text{nilai hash} = 1 \end{aligned}$$

Hasil *dot product* akan membentuk kode *hashing* dan dikelompokkan ke dalam *bucket*. Untuk semua vektor fitur citra kain tenun ikat dan batik yang mirip atau sama hasil *hashing* akan dimasukkan ke dalam *bucket* yang sama. Pengelompokan vektor fitur mempercepat proses *retrieval* yakni pengukuran kemiripan dengan metode *hamming distance* menemukan *bucket* yang sama. Pengukuran kemiripan dengan *euclidean distance* dalam *bucket* yang sama untuk mencari citra yang sesuai dengan citra *query* atau citra uji.



Gambar 5. Proses pengukuran kemiripan menggunakan metode LSH.

## 2.5 Pengukuran kinerja

Efektivitas dari suatu sistem *retrieval* citra adalah kemampuannya untuk memanggil berbagai dokumen dari suatu basis data sesuai dengan permintaan pengguna [12]. Akurasi merupakan perbandingan prediksi yang benar dibagi dengan banyaknya citra yang ditampilkan berdasarkan perangkingan. Perangkingan berdasarkan nilai jarak. Akurasi digunakan untuk mengevaluasi kinerja model yang diimplementasikan pada persamaan (2). Total citra uji menunjukkan banyaknya citra yang ditampilkan pada proses uji.

$$\text{Akurasi} = \frac{\text{jumlah citra benar}}{\text{total citra uji}} \times 100\% \quad (2)$$

Untuk menguji kemiripan citra uji dan citra dalam *database* menggunakan metode *Hamming Distance* (HD) pada persamaan (3). Pengukuran jarak antara dua *string* yang ukurannya sama dengan membandingkan simbol-simbol yang terdapat pada kedua *string* pada posisi yang sama. *Hamming distance* dari dua *string* adalah jumlah simbol dari kedua *string* yang berbeda [3].

$$\text{SIM}_H = \frac{N - (\#(B_{Q,i} \neq B_{D,i}))}{N} \quad (3)$$

dengan:

$\text{SIM}_H = \text{similarity}$

$N$  = jumlah elemen kode *hash*

$\#(B_{Q,i} \neq B_{D,i})$  = jumlah *digit* kode *hash* yang berbeda

Pengujian kemiripan antara citra uji dan citra dalam *bucket* menggunakan metode *Euclidean Distance* [13]. *Euclidean Distance* adalah metode perhitungan jarak dari dua buah titik dalam *euclidean space*. Perbandingan fitur citra dalam *database* dengan citra *query* menggunakan metode *Euclidean Distance* [14] terdapat pada persamaan (4). Metode ED banyak digunakan dalam berbagai penelitian tentang *image retrieval*.

$$d_i(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (4)$$

dengan:

$d$  = jarak *Euclidean*

$x$  = nilai vektor ciri data uji

$y$  = nilai vektor ciri data referensi atau data *query*

## 3. HASIL DAN PEMBAHASAN

Penelitian tentang sistem temu kembali citra kain tenun ikat dan batik menggunakan metode LSH dan CNN menggunakan platform *Google Collaboratory*. Spesifikasi *hardware* yang

digunakan adalah Intel (R) Xeon(R) CPU @ 2.20 GHz, T4 GPU NVIDIA-SMI, RAM 15 GB, dan *disk space* 166,8 GB. Selain itu untuk menjalankan pelatihan dan pengujian menggunakan laptop i7 *Harddisk* SSD 1 TB dan 20 GB RAM. Untuk implementasi program sistem temu kembali digunakan Program *Python* versi 3.10.12 yang memiliki *library* mencakup *machine learning* yakni *Keras* dan *TensorFlow*. Parameter *hyperparameter* yang digunakan dalam penelitian ini meliputi ukuran *batch* sebesar 64, *learning rate* 0,001, dan 30 *epoch*. Penelitian ini menggunakan *optimizer Adam* untuk memperbarui bobot jaringan secara iteratif berdasarkan fungsi kerugian yang telah ditentukan. Pemilihan parameter *hyperparameter* ini didasarkan pada penelitian yang ada yang menunjukkan efektivitasnya [9] [15].

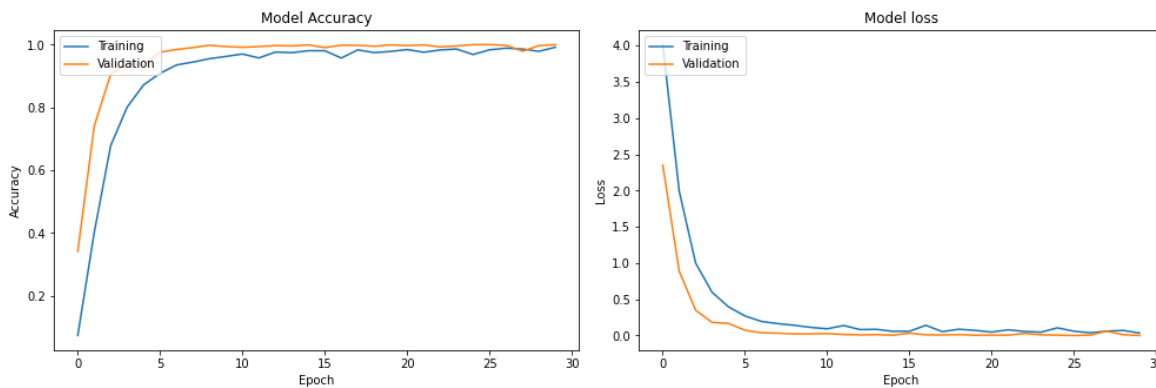
### 3.1 Hasil Eksperimen

Berdasarkan pengujian empiris bahwa metode CNN memiliki kemampuan untuk ekstraksi fitur. Metode CNN memiliki lapisan tersembunyi dan arsitektur yang lebih kompleks serta memiliki kemampuan pembelajaran fitur dan representasi fitur yang lebih canggih. Metode CNN merupakan algoritma pembelajaran mendalam (*deep learning*) yang dapat melatih kumpulan data besar dengan jutaan parameter dan mengambil bentuk gambar 2D sebagai masukan, serta menggabungkannya dengan kernel untuk menghasilkan keluaran yang diinginkan. Tabel 2 menampilkan akurasi pelatihan dan validasi untuk kedua jenis *dataset*. Pelatihan dengan *epoch* 30 menunjukkan akurasi *training* mendekati 100%. Arsitektur CNN yang dibangun disesuaikan dengan karakteristik *dataset* yang banyak mengandung fitur-fitur basis berupa garis, tepi, sudut, dan titik yang terdapat pada citra kain tenun ikat dan batik. Hasil pengujian menunjukkan bahwa pada *dataset* batik terdapat nilai *loss* validasi yang lebih kecil sebesar 0,001. Hal ini mengindikasikan bahwa model dapat mempelajari dengan baik *dataset* batik. Namun akurasi validasi yang lebih rendah dibandingkan dengan *dataset* kain tenun ikat. Hasil penelitian menunjukkan bahwa metode CNN dapat mengekstraksi fitur visual citra kain tenun ikat dan batik. Fitur hasil ekstraksi dikonversi menjadi kode *biner* baik untuk citra yang digunakan sebagai *database* maupun citra uji. Gambar 6 menunjukkan grafik pelatihan untuk akurasi dan *loss* baik untuk *training* maupun validasi. Berdasarkan grafik dapat dijelaskan bahwa kedua jenis *dataset* mengalami *overfitting* pada proses pelatihan dan validasi. Namun pada *epoch* 13 ke atas kinerja pembelajaran

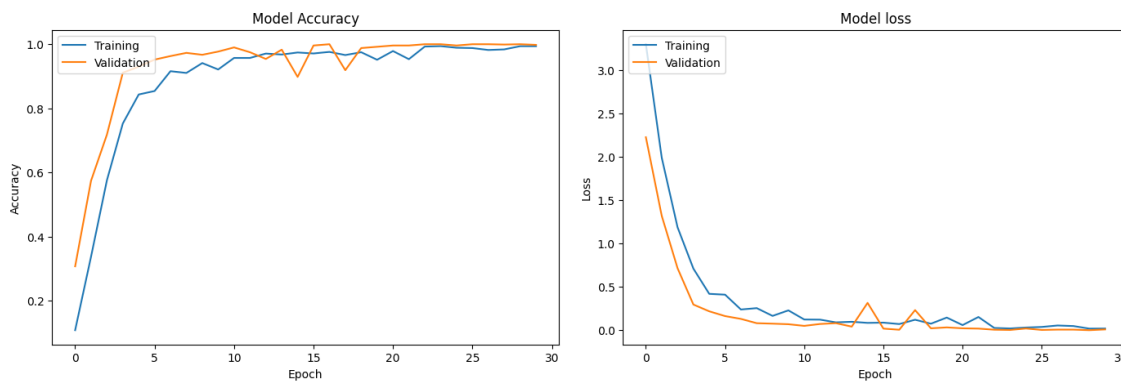
menuju konvergen mendekati 0 untuk *loss* dan 1 untuk akurasi. Hal ini terjadi karena jumlah *dataset* terbatas sehingga model tidak dapat mempelajari bobot selama proses pelatihan. Variabilitas *dataset* juga mempengaruhi pembelajaran model. Hasil pelatihan model berkinerja baik adalah nilai akurasi pelatihan dan validasi memiliki perbedaan paling kecil. Selain itu juga nilai *loss* yang paling kecil baik pelatihan maupun validasi.

**Tabel 2.** Akurasi *training* dan *validation* untuk kedua jenis *dataset*.

| <i>Dataset</i> | <i>Loss</i> | Akurasi <i>Training</i> (%) | <i>Val. Loss</i> | Akurasi <i>Val.</i> (%) |
|----------------|-------------|-----------------------------|------------------|-------------------------|
| TenunIkatNet   | 0,033       | 99,20                       | 0,004            | 100                     |
| Batik300       | 0,020       | 99,38                       | 0,001            | 99,80                   |



(a)



(b)

**Gambar 6.** Grafik akurasi dan *loss* untuk *training* dan *validation*, (a) TenunIkatNet, b) Batik300.

**Tabel 3.** Perbandingan akurasi dan waktu *retrieval* untuk kedua jenis *dataset*.

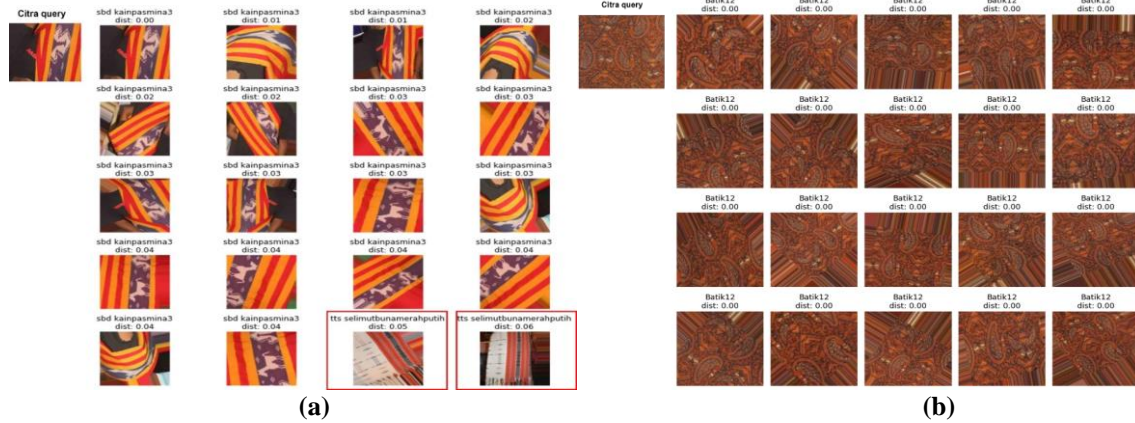
| <i>Dataset</i> | Akurasi <i>retrieval</i> rata-rata (%) |       | Waktu <i>retrieval</i> rata-rata (detik) |      |
|----------------|--|-------|--|------|
|                | LSH+HD+ED                              | ED    | LSH+HD+ED                                | ED   |
| TenunIkatNet   | 99,705                                 | 94,17 | 0,252                                    | 4,5  |
| Batik300       | 84,00                                  | 82,00 | 0,03                                     | 0,92 |

Perbandingan akurasi dan waktu *retrieval* ditampilkan pada Tabel 3. Untuk *dataset* TenunIkatNet dan Batik300 bahwa akurasi *retrieval* metode LSH+HD+ED mengungguli ED. Hal ini dipengaruhi oleh proses LSH dalam mengelompokkan citra yang mirip kedalam *bucket* yang sama. Waktu *retrieval* rata-rata metode LSH+HD+ED lebih cepat dibandingkan

dengan ED. Gambar 7 menunjukkan hasil pengukuran kemiripan citra uji dan citra dalam *dataset* baik untuk tenun ikat maupun batik. Pada *dataset* TenunIkatNet terdapat dua citra yang salah kenal dan berbeda dengan citra *query*. Berdasarkan perhitungan jarak *euclidean* yang dilakukan perangkingan, citra tersebut dikenali sebagai bagian dari kategori *sbd kainpasmina3*,

namun seharusnya citra tersebut adalah kategori *tts selimut bunamerahputih*. Pada proses perhitungan probabilitas jenis citra yang salah kenal tersebut tergabung dalam *bucket* kategori *sbd kainpasmina3*. Sehingga pada proses

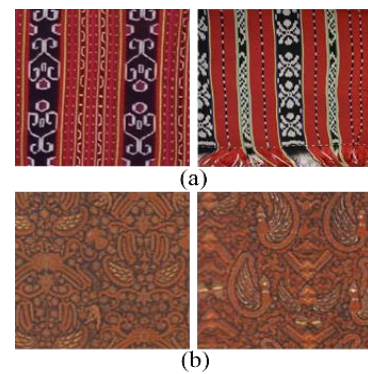
perhitungan jarak *euclidean* antara citra dikenali sebagai kategori yang sama, tetapi kenyataannya bahwa citra berasal dari kategori yang berbeda. Hal ini dapat mempengaruhi akurasi *retrieval* baik untuk citra tenun ikat maupun batik.



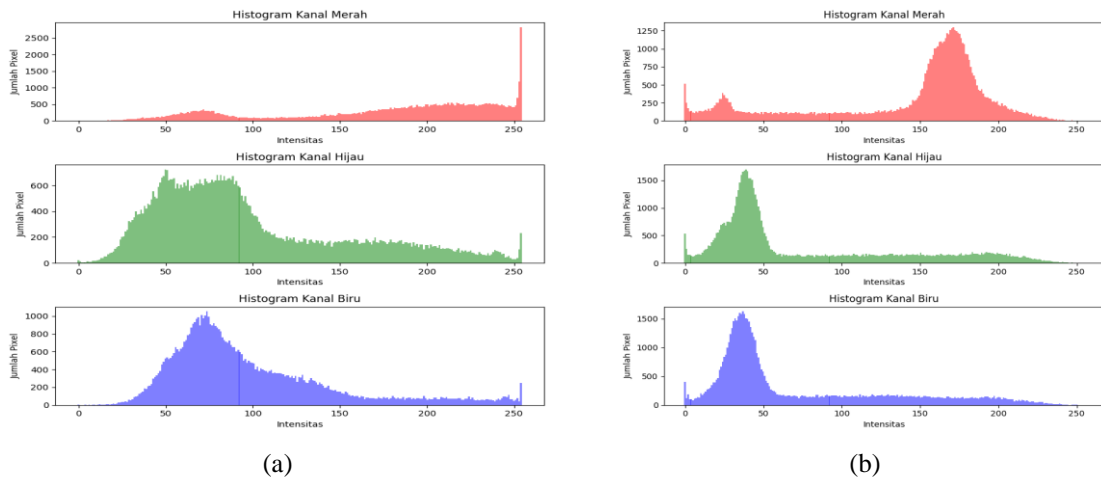
**Gambar 7.** Visualisasi hasil *retrieval* citra untuk *top-20* dan kotak merah adalah citra yang salah kenal (a) TenunIkatNet, (b) Batik300.

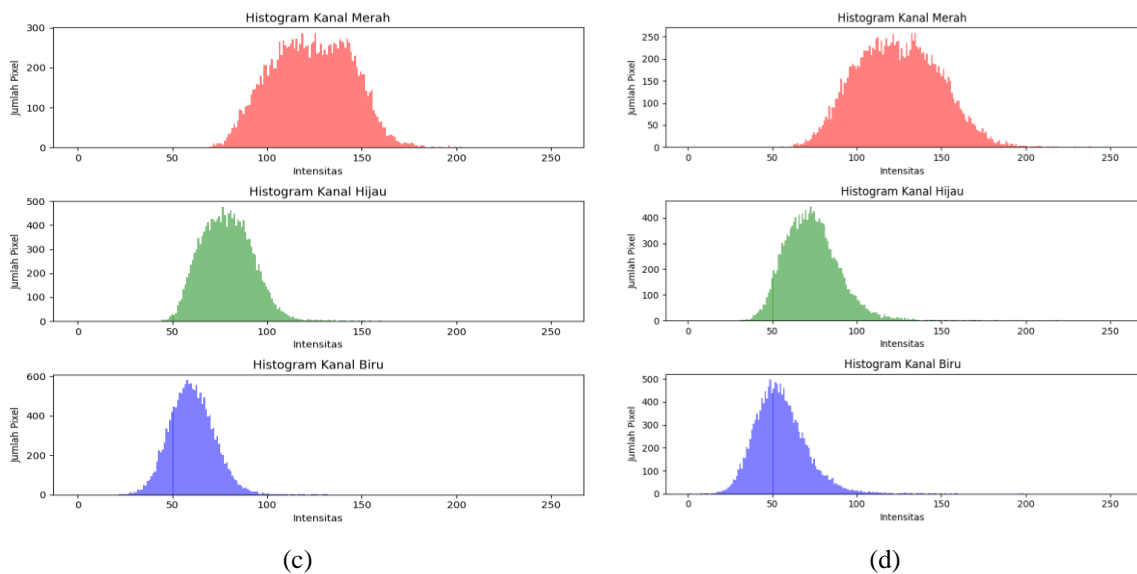
Kemiripan secara visual dapat diuji dengan perhitungan histogram untuk warna, perhitungan metrik tekstur, dan jarak berdasarkan fitur kedua citra. Gambar 8 menampilkan citra tenun ikat dan batik dari kedua kategori berbeda namun secara visual terlihat mirip. Berdasarkan fitur visual terlihat bahwa kedua jenis motif mirip yang mengakibatkan pada proses *retrieval* terjadi salah kenal dan tidak sesuai kategorinya. Pada proses *indexing* citra yang mirip dan memiliki probabilitas yang saling mendekati akan tergabung dalam satu *bucket*. Untuk membuktikan kemiripan kedua jenis citra tersebut dianalisis dengan histogram fitur warna. Gambar 9 menampilkan histogram fitur warna untuk kedua jenis citra berdasarkan kanal warna merah, hijau, dan biru. Untuk semua kanal warna terlihat bahwa ada kemiripan kedua citra tersebut

dari kategori berbeda. Nilai level keabuan pada setiap kanal warna antara 0 dan 255. Nilai level keabuan semakin tinggi seiring dengan kecerahan pada citra.



**Gambar 8.** Citra yang mirip, a) tenun ikat, b) batik.





**Gambar 9.** Histogram citra yang mirip, a) Citra tenun ikat jenis *ttu selimutbuna*, b) Citra tenun ikat jenis *belu kainselimut1*, d) Citra *batik12*, e) Citra *batik13*.

### 3.2 Pembahasan

Pelatihan model CNN untuk ekstraksi fitur citra tenun ikat dan batik masih mengalami *overfitting*. Beberapa faktor yang menyebabkan *overfitting* yakni terbatasnya data pelatihan, kualitas dan variabilitas data citra, fitur dalam data memiliki hubungan yang kuat satu sama lain (multikolinearitas) dan model yang tidak sesuai dengan karakteristik citra. Di lain pihak terdapat beberapa strategi untuk mengatasi *overfitting* pada pelatihan model antara lain augmentasi data, teknik regularisasi, dan pengaturan *hyperparameter*. Model CNN yang divariasikan dengan *padding*, teknik augmentasi data citra, dan menggunakan *dropout* juga tidak sepenuhnya menghilangkan *overfitting*. Hasil pengujian dan evaluasi model CNN pada *dataset* TenunIkatNet bahwa dari 960 citra uji terdapat 71 jenis citra kain tenun ikat NTT yang tidak mencapai akurasi *retrieval* 100% dan terdistribusi pada 44 kategori. Sedangkan *dataset* batik terdiri dari 50 kategori yakni 9 kelas yang akurasi tidak mencapai 100%. Jumlah citra uji sebanyak 32 dari 200 yang salah kenal pada proses pengukuran kemiripan. Salah satu contohnya yang ditampilkan pada Gambar 8. Beberapa jenis batik memiliki kemiripan sehingga menghasilkan akurasi *retrieval* lebih rendah dibandingkan dengan *dataset* TenunIkatNet. Implementasi metode LSH+HD+ED menghasilkan akurasi lebih baik dibandingkan pengukuran kemiripan secara langsung menggunakan metode ED. Hasil pengujian bahwa waktu *retrieval* dengan ED pada *dataset* TenunIkatNet lebih lama dibandingkan dengan *dataset* Batik300. Waktu *retrieval* seiring dengan

jumlah data citra pada *database*. Untuk citra tenun memiliki citra *database* yang jauh lebih banyak dibandingkan dengan *dataset* batik. Metode LSH dapat mengurangi dimensi vektor fitur sehingga mempercepat proses pengukuran kemiripan (*similarity measure*). Penelitian lebih lanjut adalah menemukan model CNN untuk ekstraksi fitur yang cocok dengan karakteristik *Dataset* Batik300 sehingga meningkatkan akurasi *retrieval*.

### 4. KESIMPULAN

Sistem temu kembali citra (*image retrieval*) yang cepat menjadi tantangan tersendiri dalam pengenalan citra. Meskipun perangkat keras yang digunakan sangat mendukung dan kecepatan internet, namun dalam menampilkan hasil pencarian dibutuhkan waktu yang relatif singkat. Penggunaan metode untuk pencarian menjadi salah satu bagian terpenting dalam *image retrieval*. Selain itu adalah pemilihan jenis fitur dan metode ekstraksi. Dalam penelitian ini ekstraksi fitur citra menggunakan metode *Convolutional Neural Network*. Pada proses pengukuran kemiripan menggunakan metode *Locality Sensitivity Hashing* (LSH), *Hamming Distance* dan *Euclidean Distance*. Metode LSH membentuk tabel *hash* dalam bentuk bucket yakni mengelompokkan citra yang mirip berdasarkan probabilitas. Metode LSH bertujuan mereduksi dimensi fitur citra sehingga mempercepat proses *retrieval*. Untuk proses pencarian citra uji, pengukuran awal yakni menggunakan *Hamming Distance* untuk mencari



*bucket* yang sesuai dan mengandung citra yang dicari. Ketika *bucket* yang sesuai ditemukan maka untuk mencari citra yang mirip menggunakan metode *Euclidean Distance*. Hasil penelitian menunjukkan bahwa menggunakan metode LSH+HD+ED memberikan hasil lebih cepat dibandingkan dengan ED. Waktu retrieval citra untuk LSH+HD+ED sebesar 0,21 detik, sedangkan menggunakan ED sebesar 4,5 detik untuk dataset TenunIkatNet. Sedangkan dataset Batik300 yakni 0,03 detik dan 0,9 detik. Penggunaan metode LSH sangat efektif untuk dataset yang jumlahnya besar. Metrik pengukuran akurasi menggunakan metode LSH+HD+ED berturut-turut sebesar 99,705% dan 84% untuk dataset TenunIkatNet dan Batik300. Sedangkan metode ED menghasilkan akurasi retrieval berturut-turut yakni 94,17% dan 82%.

### UCAPAN TERIMAKASIH

Penulis menyampaikan terima kasih kepada editor dan para *reviewer* yang memberikan masukan dan komentar konstruktif untuk perbaikan kualitas paper ini.

### DAFTAR PUSTAKA

- [1] Y. Luo, W. Li, X. Ma, and K. Zhang, "Image Retrieval Algorithm Based on Locality-Sensitive Hash Using Convolutional Neural Network and Attention Mechanism," *Information*, vol. 13, no. 10, p. 446, Sep. 2022, doi: 10.3390/info13100446.
- [2] J. Xiang, N. Zhang, R. Pan, and W. Gao, "Fabric retrieval based on multi-task learning," *IEEE Trans. Image Process.*, vol. 30, no. 2, pp. 1570–1582, 2021, doi: 10.1109/TIP.2020.3043877.
- [3] J. Xiang, N. Zhang, R. Pan, and W. Gao, "Fabric image retrieval system using hierarchical search based on deep convolutional neural network," *IEEE Access*, vol. 7, pp. 35405–35417, 2019, doi: 10.1109/ACCESS.2019.2898906.
- [4] J. Wang *et al.*, "Automatic Assessment of Fabric Smoothness Appearance Based on a Compact Convolutional Neural Network with Label Smoothing," *IEEE Access*, vol. 8, pp. 26966–26974, 2020, doi: 10.1109/ACCESS.2020.2971506.
- [5] N. Zhang, R. Shamey, J. Xiang, R. Pan, and W. Gao, "A novel image retrieval strategy based on transfer learning and hand-crafted features for wool fabric," *Expert Syst. Appl.*, vol. 191, no. November 2021, p. 116229, 2022, doi: 10.1016/j.eswa.2021.116229.
- [6] A. Y. Wicaksono, N. Suciati, C. Fatichah, K. Uchimura, and G. Koutaki, "Modified convolutional neural network architecture for batik motif image classification," *IPTEK J. Sci.*, vol. 2, no. 2, pp. 26–30, 2017, doi: 10.12962/j23378530.v2i2.a2846.
- [7] H. Prasetyo and B. A. Putra Akardihas, "Batik image retrieval using convolutional neural network," *TELKOMNIKA (Telecommunication Comput. Electron. Control.*, vol. 17, no. 6, p. 3010, Dec. 2019, doi: 10.12928/telkomnika.v17i6.12701.
- [8] B. Baso and N. Suciati, "Temu Kembali Citra Tenun Nusa Tenggara Timur menggunakan Ekstraksi Fitur yang Robust terhadap Perubahan Skala, Rotasi, dan Pencacayaan," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 2, p. 349, Feb. 2020, doi: 10.25126/jtiik.2020722002.
- [9] S. Tena, R. Hartanto, and I. Ardiyanto, "Content-Based Image Retrieval for Traditional Indonesian Woven Fabric Images Using a Modified Convolutional Neural Network Method," *J. Imaging*, vol. 9, no. 8, 2023, doi: 10.3390/jimaging9080165.
- [10] A. E. Minarno, M. Y. Hasanuddin, and Y. Azhar, "Batik Images Retrieval Using Pre-trained model and K-Nearest Neighbor," *Int. J. Informatics Vis.*, vol. 7, no. 1, pp. 115–121, 2023, doi: 10.30630/joiv.7.1.1299.
- [11] J. Yang and G. Yang, "Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer," *Algorithms*, vol. 11, no. 3, 2018, doi: 10.3390/a11030028.
- [12] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, "A Guide to Convolutional Neural Networks for Computer Vision," *Synth. Lect. Comput. Vis.*, vol. 8, no. 1, pp. 1–207, Feb. 2018, doi: 10.2200/S00822ED1V01Y201712COV015.
- [13] O. García-Olalla, E. Alegre, L. Fernández-Robles, E. Fidalgo, and S. Saikia, "Textile retrieval based on image content from CDC and webcam cameras in indoor environments," *Sensors (Switzerland)*, vol. 18, no. 5, pp. 1–20, 2018, doi: 10.3390/s18051329.
- [14] A. Qayyum, S. M. Anwar, M. Awais, and M. Majid, "Medical image retrieval using deep convolutional neural network," *Neurocomputing*, vol. 266, pp. 8–20, 2017, doi: 10.1016/j.neucom.2017.05.025.
- [15] M. Liu *et al.*, "Focused dropout for convolutional neural network," *Appl. Sci.*, vol. 12, no. 15, p. 7682, 2022, doi: 10.3390/app12157682.