

Klasifikasi Gambar Asli Dan Manipulasi Menggunakan Error Level Analysis (ELA) Sebagai Proses Komputasi Metode Convolutional Neural Network (CNN)

Probo Novian Candra¹, Aditya Prapanca²,

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

¹probo.18082@mhs.unesa.ac.id

²adityaprapanca@unesa.ac.id

Abstrak – Salah satu contoh penyebaran gambar palsu yang ada di media sosial dapat membuat gambar diragukan keasliannya, forensik gambar bisa menjadi alternatif untuk menguji kebenaran gambar tersebut. Umumnya mekanisme forensik gambar dapat digunakan dalam bidang studi untuk mengidentifikasi keaslian suatu gambar dari struktur kualitas gambar tersebut. Ada beberapa mekanisme yang dapat diterapkan untuk menentukan tingkat keaslian gambar, salah satunya dengan menentukan kualitas hasil tingkat kompresi gambar pada mekanisme *error level analysis* (ELA). Dalam pengembangan *image processing*, *convolutional neural network* (CNN) merupakan salah satu algoritma dari *deep learning multi layer perceptron* (MLP) dirancang sebagai pengelolah data dalam bentuk *grid* pada citra dua dimensi gambar atau suara terdiri dari *layer* utama *convolutional layer*, *polling layer*, dan *fully connected Layer* sehingga juga bisa mendeteksi atau mengklasifikasi gambar yang termodifikasi. Pada penelitian ini metode *convolutional neural network* digunakan sebagai klsifikasi data yang terlabel dengan menggunakan mekanisme *supervised learning*. Mekanisme dari *supervised learning* yaitu terdapat data yang dilatih dan terdapat variabel yang ditargetkan. Tujuan dalam penelitian ini yaitu memanfaatkan mekanisme yang dihasilkan *error level analysis* (ELA) dengan mengkonversi nilai gambar RGB ke YcrCb berformat *JPEG* untuk mencari nilai rata-rata *luminance* dan *chrominance* sehingga menghasilkan gambar *lossy*, yang mampu membantu sebagai pengenalan ekstraksi objek pada proses komputasi *learning convolutional neural network* (CNN) dengan dengan 5 *layer* utama *convolutional layer*, *polling layer*, dan 3 *fully connected layer*. Sehingga hasil dari penelitian yang didapat menghasilkan akurasi klasifikasi 97% dalam menentukan gambar asli dan manipulasi dengan nilai konverhensi mencapai 150 epoch.

Kata Kunci— Classification Image Manipulation, Error Level Analysis (ELA), Deep Learning, convolutional neural network (CNN).

I. PENDAHULUAN

Berkembangnya teknologi digital pada zaman saat ini dapat memungkinkan perangkat untuk membuat atau melakukan pencitraan gambar digital dengan resolusi tinggi

dan biaya rendah. Permasalahan tersebut menyebabkan penggunaan citra digital digunakan untuk berbagai keperluan. Dalam penggunaannya citra digital sering disalah gunakan dengan cara memanipulasi gambar untuk mengubah isi dari citra asli. Salah satu contoh penyebaran gambar palsu yang ada di media sosial dapat membuat gambar diragukan keasliannya, forensik gambar bisa menjadi alternatif untuk menguji kebenaran gambar tersebut. Umumnya mekanisme forensik gambar dapat digunakan dalam bidang studi untuk mengidentifikasi keaslian suatu gambar dari struktur kualitas gambar tersebut. Ada beberapa mekanisme yang dapat diterapkan untuk menentukan tingkat keaslian gambar. Antara lain meliputi otentikasi aktif dan otentikasi pasif. Mekanisme otentikasi aktif memerlukan sebuah informasi tambahan tentang citra asli. Sama halnya dengan proses *embedding* dan *watermarking* mekanisme tersebut juga memerlukan citra tambahan untuk mengekstraksi fitur gambar yang sudah termodifikasi. Kemudian mekanisme otentikasi pasif atau dikenal sebagai teknik deteksi anomali dalam mekanismenya tidak memerlukan informasi tambahan terkait citra asli tersebut. Pada mekanisme otentikasi pasif ada dua kategori dalam mengidentifikasi suatu gambar termanipulasi. Yaitu dengan mengidentifikasi informasi dari sumber citra gambar dan mendeteksi kontur citra gambar. Banyak cara dan mekanisme dalam mendeteksi gambar manipulasi diantaranya mengacu pada penggunaan analisis atau teknik statistik untuk mendeteksi dari gambar manipulasi tersebut.

Error level analysis (ELA) adalah salah satu teknik untuk mengetahui suatu gambar asli maupun termodifikasi dengan menghitung perbedaan rata-rata kuantitas nilai Y (*luminance*) dan CrCb (*Chorminance*) untuk menghasilkan konversi warna dominan hitam dan putih pada area gambar asli maupun manipulasi. Secara garis besar ketika gambar digital berformat *JPEG* pertama kali disimpan, maka itu merupakan gambar asli dengan kompresi format original. Akan tetapi jika gambar dilakukan pengeditan seperti perangkat lunak *adobe photoshop*, *gimp*, dan *adobe lightroom* yang mendukung operasi kompresi *JPEG*, maka kompresi format *JPEG* akan berubah dan menurunkan kualitas suatu gambar.

Convolutional neural network (CNN) adalah salah satu algoritma dari *deep learning multi layer perceptron* (MLP) dirancang untuk mengolah data dalam bentuk *grid* pada citra dua dimensi gambar atau suara terdiri dari *layer* utama *convolutional layer*, *polling layer*, dan *fully connected layer* sehingga juga bisa mendeteksi atau mengklasifikasi gambar yang termodifikasi. Pada penelitian ini metode *convolutional*

neural network digunakan untuk mengklasifikasikan data yang terlabel dengan menggunakan mekanisme *supervised learning*. Cara kerja dari *supervised learning* adalah terdapat data yang dilatih dan terdapat variabel yang ditargetkan. Sehingga tujuan dalam penelitian ini yaitu memanfaatkan mekanisme yang dihasilkan *error level analysis* (ELA) dengan mengkonversi nilai gambar RGB ke YcrCb berformat *JPEG* untuk mencari nilai rata-rata *luminance* dan *chrominance* sehingga menghasilkan gambar *lossy*, yang mampu membantu sebagai pengenalan ekstraksi objek pada proses komputasi learning *convulutional neural network* (CNN) dengan dengan 5 layer utama *convulutional layer*, *polling layer*, dan 3 *fully connected layer*. Sehingga hasil dari penelitian yang didapat menghasilkan akurasi klasifikasi 97% dalam menentukan gambar asli dan manipulasi dengan nilai konverhensi mencapai 150 epoch.

II. PENELITIAN RELEVAN

Analisis penelitian terdahulu diperlukan sebagai tolok ukur dari keberhasilan metode yang digunakan dan agar tidak terjadi plagiasi. Seperti halnya penelitian berkaitan dengan *image processing* untuk klasifikasi manipulasi gambar maupun sebagai deteksi dalam konsep *deep learning*, diantaranya diantaranya seperti yang dilakukan oleh Tajuddin Manhar Mohammed, dkk pada tahun 2018 dengan judul “*Boosting Image Forgery Detection using Resampling Features and Copy-move Analysis*”. Pada penelitian tersebut, peneliti menggambarkan teknik untuk menentukan skor manipulasi untuk gambar digital berdasarkan *resampling features* dan *copy-move features*. Hasil eksperimen menunjukkan bahwa pendekatan yang diusulkan oleh peneliti mampu meningkatkan skor AUC secara konsisten sebesar 8% -10% untuk berbagai *dataset*. (Mohammed, et al., 2018)

Sedangkan pada tahun 2018, Yan Mitha Djaksana, Abu Khalid Rivai melakukan penelitian yang berjudul “*Analisis Manipulasi Citra (Image Forgery) Menggunakan Integrasi Metode Error Level Analysis dan Block Matching*”. Pada penelitian tersebut peneliti menggunakan metode *Error Level Analysis* (ELA) yang digabungkan dengan metode *Block Matching* yang berfungsi untuk membaca serta memberikan label hasil kompresi dari metode ELA. Hasil penelitian menunjukkan bahwasannya dengan menggunakan *dataset* CASIA V2.0 mampu mendeteksi sebesar 92% *image forgery*. (Djaksana & Rivai, 2018)

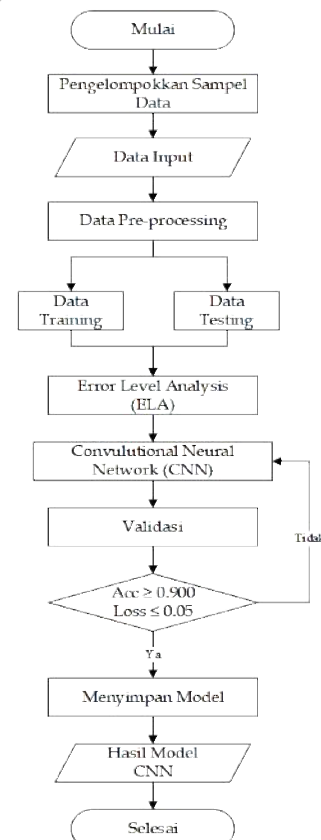
Penelitian yang dilakukan oleh Agus Gunawan, Holy Lovenia, dan Adrian Hartarto Pramudita berjudul *Deteksi Pemalsuan Gambar dengan Error Level Analysis (ELA) dan Deep Learning*. Menggunakan *Convulutional Neural Network* (CNN) dengan 2 *Convulutional Layer*, 1 *MaxPooling Layer*, 1 satu *Fully Connected Layer*, 1 *Output Layer* dengan *Softmax* dapat mencapai akurasi 91,83%, sehingga jumlah *epoch* yang dibutuhkan untuk mencapai konvergensi hanya 9. (Agus Gunawan, Holy Lovenia & Adrian Hartarto Pramudita, 2018)

Selanjutnya penelitian yang dilakukan oleh Ida Bagus Kressna Sudiattmika, Fathur Rahman, Trisno, Suyoto berjudul *Image Forgery Detection Using Error Level Analysis and Deep Learning* menggunakan model *Convulutional Neural Network*

(CNN) VGG 16-net menghasilkan akurasi 92,2% dan 88,46% validasi dengan 100 *epoch*. (Kresna Sudiattmika, Fathur Rahman, Trisno & Suyoto, 2018)

III. METODOLOGI PENELITIAN

Pada metodologi penelitian ini mempunyai beberapa batasan yang akan dicapai diantaranya memanfaatkan mekanisme yang dihasilkan *error level analysis* (ELA) dengan mengkonversi nilai gambar RGB ke YcrCb berformat *JPEG* untuk mencari nilai rata-rata *luminance* dan *chrominance* sehingga menghasilkan gambar *lossy*, yang mampu membantu sebagai pengenalan ekstraksi objek pada proses komputasi learning *convulutional neural network* (CNN) dengan dengan 5 layer utama *convulutional layer*, *polling layer*, dan 3 *fully connected layer*. Sehingga hasil dari penelitian yang didapat menghasilkan akurasi klasifikasi 97% dalam menentukan gambar asli dan manipulasi dengan nilai konverhensi mencapai 150 epoch. Data pelatihan yang digunakan dalam penelitian ini menggunakan *dataset* CASIA tahun 2018 versi 1 bersumber pada *website* www.kaggle.com. *Dataset* tersebut berupa citra gambar berformat *JPEG* terdiri atas folder gambar asli (Au) dan manipulasi (Tp). Bahasa pemrograman yang digunakan dalam menerapkan proses penelitian ini menggunakan *python* versi 3.6 serta *libarry jupyter notebook* dan *flask python*. Berikut langkah pengerjaan yang dilakukan dalam penelitian ini dijelaskan pada gambar.



Gbr. 1 Alur Proses Penelitian

A. Pengelompokan Sampel untuk Data input

Citra gambar yang digunakan dalam pelatihan proses *error level analysis* (ELA) dan *convolutional neural network* (CNN) diambil dari *dataset* CASIA tahun 2018 versi 1 bersumber pada *website* www.kaggle.com. *Dataset* yang terdiri dari 2 folder gambar asli (Au) berjumlah 7.491 dan (Tp) berjumlah 5.124 tersebut, akan dibagi menjadi beberapa nama ke dalam folder klasifikasi gambar asli dan manipulasi. kemudian untuk menyamakan ukuran data tersebut akan dilakukan penyesuaian ukuran 227×227 *pixel*. Selanjutnya *dataset* yang akan diambil untuk masing-masing kelas berjumlah 10 klasifikasi terdiri dari 5 kelas gambar asli (Au) tumbuhan, tempat, tekstur, karakter dan hewan masing-masing berjumlah 200 gambar, sehingga total dari tiap kelas gambar asli berjumlah 1.000 data. Semnata itu untuk gambar manipulasi (Tp) akan diambil 5 kelas gambar tumbuhan, tempat, tekstur, karakter dan hewan dimana masing-masing berjumlah 200 gambar, sehingga total tiap kelas dari gambar manipulasi berjumlah 1.000 data dengan total keseluruhan *dataset* klasifikasi berjumlah 2.000 data gambar asli dan manipulasi. *Dataset* dari 10 klasifikasi akan dibagi menjadi 2 jenis kelompok data diantaranya proses *training* sebesar 90% dan untuk proses *testing* atau pengujian sebesar 10% dari total keseluruhan *dataset* yang akan digunakan dalam pelatihan proses *error level analysis* (ELA) dan *convolutional neural network* (CNN). Berikut beberapa contoh kelas gambar asli dan manipulasi:



Gbr. 2 Tumbuhan Asli



Gbr. 3 Tumbuhan Manipulasi

Penjelasan gambar 2 merupakan kelas gambar tumbuhan asli karena tidak ada tambahan objek **daun**. Berbeda dengan gambar 3 merupakan kelas gambar tumbuhan manipulasi karena terdapat tambahan objek **daun** pada tumbuhan tersebut.



Gbr. 5 Tempat Asli

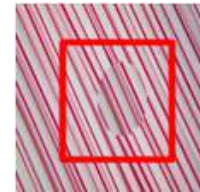


Gbr. 4 Tempat Manipulasi

Penjelasan gambar 4 merupakan kelas gambar tempat asli karena tidak ada tambahan objek **simbol bintang**. Berbeda dengan gambar 5 merupakan kelas gambar tempat manipulasi karena terdapat tambahan objek **simbol bintang** pada tempat tersebut.



Gbr. 7 Tekstur Asli



Gbr. 6 Tekstur Manipulasi

Penjelasan gambar 6 merupakan kelas gambar tekstur asli karena tidak ada tambahan objek **efek blur** dibagian tengah. Berbeda dengan gambar 7 merupakan kelas gambar tekstur manipulasi karena terdapat tambahan objek **efek blur** dibagian tengah pada tekstur tersebut.



Gbr. 8 Hewan Asli



Gbr. 9 Hewan Manipulasi

Penjelasan gambar 8 merupakan kelas gambar hewan asli karena tidak ada tambahan objek **rumpun** dibagian bawah. Berbeda dengan gambar 9 merupakan kelas gambar hewan manipulasi karena terdapat tambahan objek **rumpun** dibagian bawah pada tekstur tersebut.



Gbr. 11 karakter Asli



Gbr. 10 karakter Manipulasi

Penjelasan gambar 10 merupakan kelas gambar karakter asli karena tidak ada tambahan objek **anak kecil** dibagian bawah. Berbeda dengan gambar 11 merupakan kelas gambar karakter manipulasi karena terdapat tambahan objek **anak kecil** dibagian bawah pada tekstur tersebut.

B. Error Level Analysis (ELA)

Selanjutnya menerapkan *error level analysis* (ELA) dengan mengkonversi nilai gambar *RGB* ke *YcrCb* berformat *JPEG* untuk mencari nilai rata-rata *Y luminance* dan *CrCb chrominance* sehingga menghasilkan nilai *lossy* pada setiap karakteristik gambar asli maupun manipulasi, berikut mekanisme *error level analysis* (ELA). (I Gede Nengah Bayu Darmawan, Gusti Made Arya Sasmitha, Putu Wira Buana, 2019). diantaranya:

1. Konversi nilai *RGB* ke *YcrCb*

Dengan menentukan nilai *luminance* dan *crominance* pada *kernel 8x8* citra kompresi dengan rumus, Sebagai contoh :

Rumus

$$Y = 0.257 * R + 0.504 * G + 0.098 * B + 16$$

$$Cb = -0.257 * R - 0.291 * G + 0.439 * B + 128$$

$$Cr = 0.439 * R - 0.368 * G - 0.071 * B + 128$$

Contoh Perhitungan

$$Y = 0.257 * 2 + 0.504 * 1 + 0.098 * 3 + 16$$

$$= 0.5 + 0.5 + 0.3 + 16 = 17.3$$

$$Cb = -0.257 * 2 - 0.291 * 1 + 0.439 * 3 + 128$$

$$= -0.3 - 0.3 + 1.3 + 128 = 128.7$$

$$Cr = 0.439 * 8 - 0.368 * 1 - 0.071 * 3 + 128$$

$$= 0.9 - 0.4 - 0.2 + 128 = 128$$

2. Menghitung nilai Error Level Analysis (ELA)

Langkah selanjutnya setelah mengkonversi nilai RGB ke YcrCb yaitu mencari perbedaan nilai rata-rata dari Luminance dan Crominance (μ) yang menghasilkan nilai Error Level Analysis (ELA) (Qn).

$$\mu = \frac{Y + Cr + Cb}{3}$$

$$\Delta = |Y - Cr| \times (1.0 - 0.51) + |Y - Cb| \times (1.0 - 0.51)$$

$$Qn = 100 - \mu - \Delta$$

Contoh Perhitungan :

$$\mu = \frac{17 + 129 + 128}{3}$$

$$\Delta = |17 - 129| \times (1.0 - 0.51) + |17 - 128| \times (1.0 - 0.51)$$

$$= 112 \times (1.0 - 0.51) + 111 \times (1.0 - 0.51)$$

$$= 54.9 + 54.3 = 109.3$$

$$Qn = 100 - \mu - \Delta = 100 - 91.3 - 109.3 = -100.6$$

Contoh Perhitungan :

$$\mu = \frac{17 + 129 + 128}{3}$$

$$\Delta = |17 - 129| \times (1.0 - 0.51) + |17 - 128| \times (1.0 - 0.51)$$

$$= 112 \times (1.0 - 0.51) + 111 \times (1.0 - 0.51)$$

$$= 54.9 + 54.3 = 109.3$$

$$Qn = 100 - \mu - \Delta = 100 - 91.3 - 109.3 = -100.6$$

Perhitungan nilai error level analysis (ELA) akan dilakukan dari kordinat pixel (1,1) sampai dengan kordinat pixel (8,8). Perhitungan pada grid 8x8 pixel tersebut, akan terus dilakukan hingga panjang atau lebar pixel pada citra kurang dari 8 pixel. Setelah dilakukan perhitungan terhadap nilai error level analysis (ELA) pada grid 8 x 8 pixel, proses tersebut akan melakukan pemilahan nilai yang berbeda. Jika nilai tersebut berbeda maka dianggap sebagai nilai lossy, dan jika nilai tersebut melewati nilai ambang batas yang ditentukan misalnya nilai yang ditentukan adalah 1/4 dari

jumlah blok pixel (64) maka variabel temp akan ditambahkan nilai 1.

Variabel temp merupakan variabel bantu untuk menyimpan hasil perhitung per grid 8x8 jika grid dinilai memiliki nilai error yang melebihi ambang batas yang ditentukan maka variabel temp akan ditambahkan nilai 1 jika tidak maka penambahan nilai adalah 0.

Tabel 1. Contoh Nilai ELA per 8x8 pixel

81	81	81	81	81	81	81	81
81	81	81	81	81	81	81	81
81	81	81	81	81	81	81	81
90	90	90	90	81	81	81	81
90	90	90	90	81	81	81	81
90	90	90	90	81	81	81	81
90	90	90	90	81	81	81	81
90	90	90	90	81	81	81	81



Gbr. 12 Gambar Asli ELA



Gbr. 13 Gambar Asli Konversi ELA



Gbr. 14 Input Gambar Manipulasi ELA



Gbr. 15 Hasil Gambar Manipulasi Konversi ELA

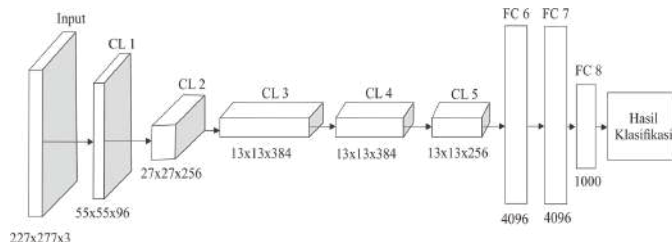
Penjelasan pada gambar 12 merupakan contoh kelas gambar tempat yang dikonversi error level analysis (ELA) dengan quality mencapai 90%. Dari hasil gambar 13 menunjukkan gambar cenderung berbentik putih artinya gambar tersebut memiliki nilai lossy yang rendah. Sedangkan gambar 17 manipulasi merupakan hasil konversi error level analysis (ELA) dengan quality mencapai 90%. Hasil gambar 15 menunjukkan gambar cenderung kontras ditandai warna kemerahan pada bagian yang dimanipulasi, artinya gambar tersebut memiliki nilai lossy yang tinggi.

Dari mekanisme yang sudah dijelaskan tersebut, hasil konversi berupa gambar lossy yang dihasilkan oleh error level analysis (ELA) akan dimasukkan ketahap proses komputasi convulutional neural network (CNN) menggunakan 5 layer utama convulutional layer, polling layer, dan 3 fully connected

layer. Dengan mekanisme tersebut diharapkan mekanisme *error level analysis* (ELA) mampu membuat pengenalan ekstraksi objek model *convolutional neural network* (CNN). Sehingga hasil dari penelitian yang didapat menghasilkan akurasi klasifikasi 97% dalam menentukan gambar asli dan manipulasi dengan nilai konverhensi mencapai 150 epoch.

C. Convolutional Neural Network (CNN)

Umumnya layer pada *convolutional neural network* (CNN) mempunyai beberapa komposisi yang baik, diantaranya pada layer pertama *convolutional layer* berisi satu set filter input 3D dengan nilai ukuran $PW \times PH \times D$ dan $QW \times QH \times D$ sehingga menghasilkan *output* aktivasi $SW \times SH$. Nilai SW dan SH dapat diperoleh dengan rumus sebagai berikut:



Gbr. 16 Arsitektur Model *Convolutional Neural Network* (CNN)

$$Sw = 1 + \frac{PW - QW + 2\beta}{\mu}$$

$$Sh = 1 + \frac{PH - QH + 2\beta}{\mu}$$

dimana μ merupakan ukuran *stride* dan β mewakili batas garis. Kemudian satu filter tersebut akan menghasilkan satu *feature map* 2D dengan ukuran $SW \times SH \times T$.

1. Convolution layer

Convolution layer merupakan bagian penting dalam CNN. Layer ini memiliki satu set filter atau *kernel* yang akan dikonvolusi dengan data *input*. Tujuan dari layer ini adalah untuk menghasilkan representasi fitur dari data input (Guo, et al., 2017). Hasil dari konvolusi disebut dengan *feature map*.

$$Y_j^n = f \left(\sum_{i=1}^{M^{n-1}} X_i^{n-1} * W_{ij}^n + b_j^n \right)$$

Dimana:

Y = hasil konvolusi / *feature map*

X = input

b = bias

2. Fungsi aktivasi

Fungsi aktivasi berfungsi untuk mengenalkan *non-linearity* kepada jaringan dengan mengubah bilangan *input* menjadi bilangan dengan *range* yang lebih kecil. Fungsi aktivasi tersebut menggunakan aktivasi *ReLU* sebagai berikut:

$$ReLU(x) = \max(0, x)$$

3. Pooling

proses *pooling* akan meringkas informasi pada sebagian data *input* (*local region*), sehingga didapatkan fitur yang lebih kecil. Proses *pooling* mempunyai dua mekanisme *pooling* diantaranya *max pooling* dan *average pooling* (Mohamed, et al., 2019). Sebagai contoh jika terdapat

sebuah *input* dengan ukuran 4×4 yang di-*pooling* dengan ukuran 2×2 dan *stride* sebesar 2. Seperti halnya konvolusi, diperlukan adanya inisialisasi ukuran *pooling* dan *stride*. Kemudian jika ukuran *pooling* adalah $f \times f$ dengan *stride* s , maka ukuran *output*-nya bisa dihitung dengan rumus:

$$h' = \left\lfloor \frac{h - f + s}{s} \right\rfloor, w' = \left\lfloor \frac{w - f + s}{s} \right\rfloor$$

4. Flatten Layer

Flatten layer merupakan layer yang bertujuan untuk mengubah data dua dimensi menjadi satu dimensi. Pada proses ini, setiap baris pada *feature map* akan disusun secara vertikal

5. Fully Connected Layer

Merupakan proses tahap klasifikasi *multilayer perceptron* (MLP) atau juga dikenal dengan *neural networks*. Proses *multilayer perceptron* (MLP) yang digunakan adalah *backpropagation*.

6. Softmax Layer

Umumnya lapisan *softmax layer* menggunakan generalisasi *multiclass* dari *regresi* jumlah data yang digunakan atau dikenal sebagai fungsi *softmax* yang terhubung setelah *fully connected*. berikut parameter model *convolutional neural network* (CNN) yang digunakan:

Tabel 2. Parameter *Convolutional Neural Network* (CNN)

Layer	Volume	Parameter
Input	227, 227 x 3	0
conv2d_1 (Conv2D)	55, 55, 96	34944
max_pooling2d_1 (MaxPooling2)	27, 27, 96	0
batch_normalization_1	27, 27, 96	384
conv2d_2 (Conv2D)	17, 17, 256	2973952
max_pooling2d_2 (MaxPooling2)	8, 8, 256	0

batch_normalization_2	8, 8, 256	1024
conv2d_3 (Conv2D)	6, 6, 384	885120
batch_normalization_3	6, 6, 384	1536
conv2d_4 (Conv2D)	4, 4, 384	1327488
batch_normalization_4	4, 4, 384	1536
conv2d_5 (Conv2D)	1, 1, 256	884992
batch_normalization_5	1, 1, 256	1024
flatten_1 (Flatten)	256	0
dense_1 (Dense)	4096	1052672
dropout_1 (Dropout)	4096	0
batch_normalization_6	4096	16384
dense_2 (Dense)	4096	16781312
dropout_2 (Dropout)	4096	0
batch_normalization_7	4096	16384
dense_3 (Dense)	1000	4097000
dropout_3 (Dropout)	1000	0
batch_normalization_8	1000	4000
dense_4 (Dense)	10	10010
Total params: 28,089,762		
Trainable params: 28,068,626		
Non-trainable params: 21,136		

D. Learning

Dalam penelitian ini, model *compile* yang digunakan untuk melakukan optimasi pada *convolutional neural network* (CNN) yaitu menggunakan *stochastic gradient descent* (SGD). Pada dasarnya mekanisme melatih sebuah model akan membutuhkan sebuah fungsi *loss function* yang dapat memungkinkan untuk mengukur suatu kualitas dari setiap bobot dan parameter tertentu. Tujuan dari pengoptimalan tersebut sebagai parameter untuk meminimalkan *loss function*. Mekanisme *Stochastic gradient descent* (SGD) akan bekerja ketika meminimalkan fungsi $J(\theta)$ yang memiliki parameter θ dengan memperbarui parameter ke suatu arah menurun. *Stochastic gradient descent* (SGD) memiliki *learning rate* (η) digunakan sebagai langkah menentukan titik minimum. *Stochastic gradient descent* (SGD) merupakan metode *gradient descent* yang melakukan *update* parameter untuk setiap data pelatihan $x(i)$ serta label $y(i)$ memiliki persamaan sebagai berikut, Shiv Ram Dubey., Snehasis Mukherjee 2020)

$$\theta_{t+1,i} = \theta_{t,i} - \eta \times g_{t,i}$$

dimana $\theta_{t,i}$ dan $\theta_{t+1,i}$ adalah nilai sebelumnya yang akan diperbarui untuk parameter ke- i dengan $i = 1, 2, \dots, d$, dimana d adalah jumlah parameter, dan $g_{t,i}$ adalah gradien dengan parameter $\theta_{t,i}$ sebagai fungsi kerugian nilai \mathcal{L} , didefinisikan sebagai berikut:

$$g_{t,i} = \frac{\partial(\mathcal{L}_t, \theta)}{\partial(\theta_{t,i})}$$

dimana \mathcal{L}_t, θ adalah fungsi kerugian parameter jaringan (θ) dalam iterasi tersebut. Dalam penelitian ini, *categorical cross entropy* digunakan untuk proses klasifikasi menggunakan fungsi aktivasi *softmax*, didefinisikan sebagai berikut:

$$\mathcal{L}_t, \theta = \frac{1}{N_b} \sum_{j=1}^{N_b} \mathcal{L}_t, \theta, j + \sigma R_{t, \theta},$$

dimana N_b adalah jumlah gambar pelatihan dalam *batch*, \mathcal{L}_t, θ , sedangkan j untuk menghitung *categorical cross entropy* di iterasi t , kemudian pada fungsi $R_{t, \theta}$ adalah menghitung fungsi kerugian *loss* dalam iterasi t , dan σ . Hilangnya kerugian pada *categorical cross entropy* pada fungsi \mathcal{L}_t, θ, j didefinisikan sebagai berikut:

$$\mathcal{L}_t, \theta, j = -\log \left(\frac{e^{S_{oj}}}{\sum_{k=1}^{N_c} e^{S_k}} \right)$$

dimana N_c adalah jumlah total kelas dalam *dataset*, sedangkan o_j adalah nilai kelas aktual kebenaran dasar dengan nilai j dan S_k adalah skor kelas yang dihitung pada kerugian $R_{t, \theta}$ dihitung sebagai berikut:

$$R_{t, \theta} = \sum_{i=1}^d (\theta_{t,i})^2$$

Compile learning menggunakan 150 *epoch*, secara mekanisme *epoch* merupakan nilai *index* dari iterasi tingkat akurasi yang dicapai. Untuk menentukan variabel maka target pada proses learning ini adalah akurasi lebih dari > 0.900 dan $loss < 0.05$. jika model memenuhi target diatas 90% maka model tersebut akan disimpan sebagai hasil terbaik dengan format H5 yang sudah ditentukan sebelumnya.

E. Penghitungan Akurasi

Variabel dalam penelitian ini adalah penghitungan akurasi model *convolutional neural network* (CNN) yang sudah dibuat untuk mengetahui klasifikasi gambar asli dan manipulasi, berikut rumus penghitungan nilai akurasi :

$$\text{Akurasi} = \frac{\text{jumlah data benar}}{\text{jumlah keseluruhan data}} \times 100\%$$

IV. HASIL DAN PEMBAHASAN

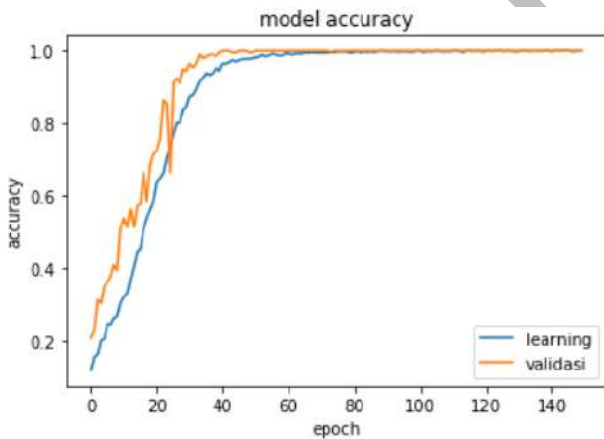
Tahap terakhir dari penelitian ini yaitu proses menguji coba mekenasim serta hasil dari proses penelitian tersebut. Proses tersebut memiliki mekanisme diantaranya menguji akurasi klasifikasi dengan mengetahui nilai indeks yang telah disusun dan oleh model *convolutional neural network* (CNN) dari model terbaik yang disimpan dalam format H5. Berikut alur dari

proses uji coba: Program yang digunakan dalam uji coba yaitu menggunakan *libarry flask python* berupa tampilan *UI* pada *web localhost*. Mekanismenya, model akurasi *convulutional neural network* (CNN) yang sudah di simpan dengan format *H5*, nantinya akan diunggah menggunakan *libarry OpenCv python* sebagai pembaca file akurasi yang telah di capai selama proses *learning* dengan mengubah ukuran *227x 227 pixel*. Selanjutnya proses klasifikasi akan menampilkan hasil pada halaman *UI* pada *web localhost*.

A. Pelatihan CNN

Dalam penelitian ini model *compile convulutional neural network* (CNN) yang digunakan menggunakan *optimizer stochastic gradient descent* (SGD) dengan perhitungan *lr=0.001, momentum=0.9 decay=0.005* dengan target *metrics accuracy*. Sementara itu pelatihan *convulutional neural network* (CNN) menggunakan batas konverhensi *150 epoch* dengan menggunakan *avarge pooling*. Kemudian fungsi untuk *loss* yang dipakai yaitu *categorical crossentropy* dengan fungsi aktivasi yang digunakan pada setiap lapisan berbeda. Fungsi *ReLu* akan diletakkan pada lapisan konvolusi sedangkan *sigmoid* digunakan untuk lapisan tersembunyi pada *fully connected layer*, dan *output layer* menggunakan *softmax*.

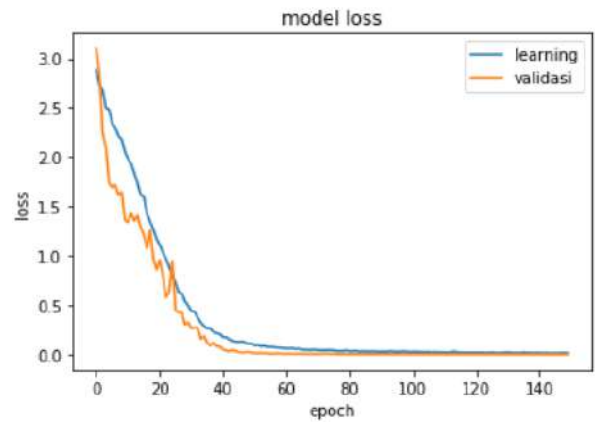
Mekanisme akurasi dalam penelitian ini yaitu mengetahui hasil klasifikasi gambar asli dan manipulasi dari fungsi *softmax* dengan probabilitas pada *output* mulai 0 hingga 1. Akurasi yang didapatkan dengan penggunaan nilai *dropout* sebesar 0,4 pada setiap lapisan setelah *pooling*, dan 0,2 pada lapisan tersembunyi di *fully conected layer*. Sementara itu, *Batch Normalization* diletakkan pada setiap lapisan setelah lapisan konvolusi. Pada grafik tersebut menunjukkan garis warna biru merupakan *training accuracy* mendapat nilai diatas 90% sedangkan garis warna kuning merupakan *validation accuracy* mendapat nilai setara dengan *training accuracy* dengan rentan 0,20% - 0,98%.



Gbr. 17 Grafik Training Accuracy dan Validation Accuracy pada proses Learning

Berikut merupakan gambar grafik *loss convulutional neural network* (CNN) yang didapatkan selama proses pelatihan data. Grafik tersebut menunjukkan garis warna

biru merupakan *training loss* mencapai nilai penurunan *loss* mulai 3,0% dengan batas *150 epoch*. Sedangkan garis warna kuning merupakan *validation loss* mencapai penurunan nilai *loss* setara dengan *training loss* dengan rentan 3,0% - 0% sampai batas *150 epoch*.









Gbr. 18 Grafik Loss Proses Learning

B. Data Uji coba

Data yang digunakan merupakan data gambar *copy-move* yaitu gambar asli yang ditambahkan objek tertentu. Sehingga dari gambar manipulasi akan terlihat objek yang tidak realistis pada gambar asli. Kemudian data *testing* yang akan digunakan sebagai uji coba pada 5 kelas gambar asli (Au) meliputi tumbuhan, tempat, tekstur, karakter dan hewan masing-masing berjumlah 20 gambar sehingga terdapat 100 gambar asli (Au). Berikut merupakan contoh data testing gambar asli (Au) pada tabel 3:


Tabel 3 Contoh Data Uji Coba Gambar Asli





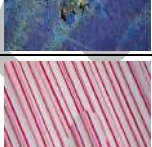



No	Gambar	Klasifikasi gambar	Resolusi	Format
1.		Tumbuhan (Asli)	227 x 227	jpg
2.		Tumbuhan (Asli)	227 x 227	jpg
3.		Tempat (Asli)	227 x 227	jpg

No	Gambar	Klasifikasi gambar	Resolusi	Format
4.		Tempat (Asli)	227 x 227	jpg
5.		Tekstur (Asli)	227 x 227	jpg
6.		Tekstur (Asli)	227 x 227	jpg
7.		Hewan (Asli)	227 x 227	jpg
8.		Hewan (Asli)	227 x 227	jpg
9.		Karakter (Asli)	227 x 227	jpg

Sama halnya dengan data *testing* gambar asli (Au), data *testing* pada gambar manipulasi (Tp) akan dibagi menjadi 5 kelas gambar manipulasi (Tp) meliputi tumbuhan, tempat, tekstur, karakter dan hewan masing-masing berjumlah 20 gambar sehingga terdapat 100 gambar manipulasi (Tp). Berikut merupakan contoh data *testing* gambar manipulasi (Tp) pada tabel 4:

Tabel 4 Contoh Data Uji Coba Gambar Manipulasi

No	Gambar	Klasifikasi gambar	Resolusi	Format
1.		Tumbuhan (Manipulasi)	227 x 227	jpg

No	Gambar	Klasifikasi gambar	Resolusi	Format
2.		Tumbuhan (Manipulasi)	227 x 227	jpg
3.		Tempat (Manipulasi)	227 x 227	jpg
4.		Tempat (Manipulasi)	227 x 227	jpg
5.		Tekstur (Tekstur)	227 x 227	jpg
6.		Tekstur (Tekstur)	227 x 227	jpg
7.		Hewan (Manipulasi)	227 x 227	jpg
8.		Hewan (Manipulasi)	227 x 227	jpg
9.		Karakter (Manipulasi)	227 x 227	jpg

C. Hasil Proses Testing

Dalam penelitian ini Variabel untuk menentukan hasil klasifikasi gambar asli dan manipulasi pada proses *testing* yaitu variabel (nilai 'benar' dan 'salah') dari klasifikasi gambar asli dan manipulasi selanjutnya variabel hasil yang didapat dari *testing* tersebut (nilai 'benar' dan 'salah') akan digunakan untuk menghitung akurasi dari model *convolution neural*

network (CNN). Berikut data yang digunakan pada proses testing:

Tabel 5 Klasifikasi Gambar Asli

No.	Gambar	Klasifikasi	Jumlah Data Uji	Salah	Benar
1.	Tumbuhan	Asli	20	0	20
2.	Tempat	Asli	20	0	20
3.	Tekstur	Asli	20	0	20
4.	Hewan	Asli	20	0	20
5.	Karakter	Asli	20	0	20
			100	0	100

Tabel 6 Klasifikasi Gambar Manipulasi

No.	Gambar	Klasifikasi	Jumlah Data Uji	Salah	Benar
1.	Tumbuhan	Manipulasi	20	1	19
2.	Tempat	Manipulasi	20	1	19
3.	Tekstur	Manipulasi	20	2	18
4.	Hewan	Manipulasi	20	0	20
5.	Karakter	Manipulasi	20	2	18
			100	6	94

Hasil *testing* yang didapat dari tabel gambar asli dari 100 data uji semuanya benar, sedangkan tabel gambar manipulasi dari 100 data uji mempunyai 6 salah dengan rincian gambar tumbuhan mempunyai kesalahan 1, gambar tempat 1, gambar tekstur 2, dan gambar karakter 2. Selanjutnya yaitu menghitung akurasi dengan penghitungan jumlah data benar dibagi jumlah keseluruhan data uji dikali 100% sebagai berikut:

$$Akurasi = \frac{194}{200} \times 100\% = 97\%$$

V. KESIMPULAN

Berdasarkan hasil uji coba penelitian mengenai Klasifikasi gambar asli dan manipulasi menggunakan *error level analysis* (ELA) sebagai proses komputasi metode *convolution neural network* (CNN), Kesimpulan tersebut adalah sebagai berikut :

Mekanisme yang dihasilkan *error level analysis* (ELA) dengan mengkonversi nilai gambar RGB ke YrCb berformat JPEG untuk mencari nilai rata-rata *luminance* dan *chrominance* menghasilkan gambar *lossy* mampu membantu sebagai pengenalan ekstraksi objek pada proses komputasi

learning convolutional neural network (CNN) dengan 5 layer utama *convolutional layer*, *polling layer*, dan 3 *fully connected layer* mendapat akurasi yang tinggi. Hal itu dibuktikan dengan tercapainya model H5 yang tersimpan secara otomatis jika mencapai akurasi diatas 90%. Model CNN yang tercapai dibuat sebagai data *testing* untuk melihat kesalahan dari klasifikasi yang dihasilkan dengan menguji 10% data *testing* dari total 2.000 gambar asli dan manipulasi. sehingga jumlah total data *testing* yang digunakan 200 gambar. Dari 200 gambar data uji tersebut terdapat kesalahan dalam menilai klasifikasi gambar manipulasi sebanyak 6 kesalahan gambar, sehingga diperoleh perhitungan dari total data *testing* yang benar sebanyak 194 gambar dibagi total keseluruhan data *testing* sebanyak 200 gambar kemudian dikali 100% sehingga mendapat tingkat akurasi dalam menentukan gambar asli dan manipulasi mencapai 97%.

VI. UCAPAN TERIMA KASIH

Puji syukur kepada Allah SWT dimana telah mempermudah dalam menyelesaikan penelitian ini. Sehingga penelitian ini dapat berjalan dan terselesaikan dengan baik. Serta terimakasih kepada semua pihak terkait dalam memberikan dukungan selama penelitian ini terselesaikan.

VII. REFERENSI

- [1] Dang, L. M., Hasan, S. I., Im, S., & Moon, H. (2019). Face image manipulation detection based on a convolutional neural network. *Expert Systems With Applications*, 156-168.
- [2] Djaksana, Y. M., & Rivai, A. K. (2018). Analisis Manipulasi Citra (Image Forgery) Menggunakan Integrasi Metode Error Level Analysis Dan Block Matching... *Jurnal Teknologi Informasi ESIT*.
- [3] Mohammed, T. M., Bunk, J., Nataraj, L., Bappy, J. H., Flenner, A., Manjunath, B. S., Peterson, L. (2018). Boosting Image Forgery Detection using Resampling Features and Copy-move Analysis. *arXiv*.
- [4] Riadi, I., Yudhana, A., & Sulistyono, W. Y. (2019). Analisis Image Forensics Untuk Mendeteksi Pemalsuan Foto Digital. *Jurnal Mobile and Forensics (MF)*, 13-21.
- [5] Ida Bagus Kresna, S., Fathur Rahman., Trisno., Suyoto. (2019). Image Forgery Detection Using Error Level Analysis and Deep Learning. *TELKOMNIKA, Vol.17, No.2*, 653-659.
- [6] N. Krawetz., "A pictures worth digital image analysis and forensics," *Black Hat Briefings*, pp. 1-31, 2007.
- [7] Gunawan, T. S., Hanafiah, S. A. M., Kartiwi, M., Ismail, N., Za'bah, N. F., & Nordin, A. N. (2017). Development of photo forensics algorithm by detecting photoshop manipulation using error level analysis. *Indonesian Journal of Electrical Engineering and Computer*
- [8] I Gede Nengah Bayu Darmawan, Gusti Made Arya Sasmitha, Putu Wira Buana., (2019). Pengembangan Metode Pendeteksi Modifikasi Citra Menggunakan Metode Error Level Analysis. *MERPATI VOL., 7, NO. 1 APRIL 2019*. 29-36
- [9] Guo, T., Dong, J., Li, H. & Gao, Y., 2017. Simple Convolutional Neural Network on Image Classification. Beijing, IEEE.

- [10] Mohamed, O., Khalid, E. A., Mohamed., O. & Brahim., A., 2019. Content-Based Image Retrieval Using Convolutional Neural Networks. pp. 463-476.
- [11] Shiv Ram Dubey, Soumendu Chakraborty, Swalpa Kumar Roy, Snehasis Mukherjee 2020. diffGrad: An Optimization Method for Convolutional Neural Networks. arXiv:1909.11015v3 [cs.LG] 6 Mar 2020.

Article In Press