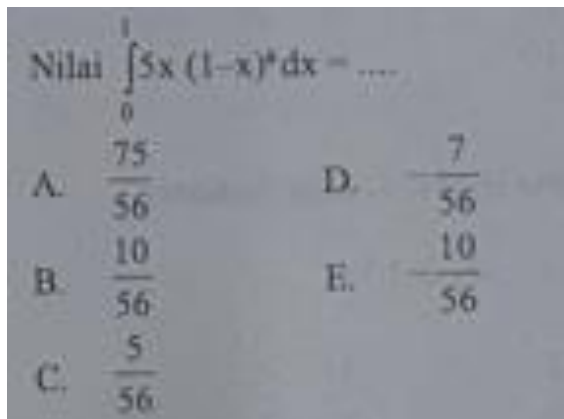


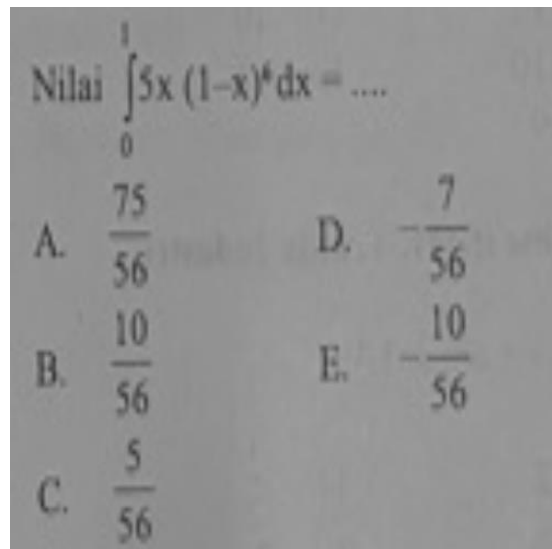
Pada penelitian ini dilakukan klasifikasi label soal integral dan peluang menggunakan metode *Convolutional Neural Network* (CNN). Kebaikan hasil klasifikasi pada penelitian ini akan diukur menggunakan *accuracy*. Penelitian ini akan menarik kesimpulan berupa jumlah *convolutional layer* terbaik yang digunakan dalam klasifikasi label soal.

1.1 Preprocessing Data Foto Soal

Pada penelitian ini, tahap *preprocessing* yang dilakukan adalah mengubah warna pada fitur ekstraksi yang digunakan menjadi berwarna hitam putih. Berikut merupakan salah satu contoh fitur ekstraksi yang sudah diubah menjadi berwarna hitam putih.



(a)



(b)

Gambar 4.1 (a) Foto soal sebelum preprocessing (b) Foto soal setelah preprocessing

Pada proses *preprocessing*, foto soal diubah menjadi berwarna hitam putih dan ukuran pixel diubah menjadi 500x500. Tahap *preprocessing* dilakukan pada semua data foto yang digunakan sehingga data input yang didapatkan sebesar 80000x500 fitur dengan 2 label, yaitu label integral dan peluang. Berikut merupakan matriks data ke-1 yang sudah diubah dari berwarna menjadi hitam putih.

$$\begin{pmatrix} 254 & 253 & 255 \\ \vdots & \vdots & \vdots \\ 251 & 253 & 254 \\ 254 & 254 & 255 \\ \vdots & \vdots & \vdots \\ 251 & 253 & 254 \\ 253 & 252 & 254 \\ \vdots & \vdots & \vdots \\ 251 & 253 & 254 \end{pmatrix}$$

(a)

$$\begin{pmatrix} 254 & 251 & 253 \\ \vdots & \vdots & \vdots \\ 253 & 253 & 253 \end{pmatrix}$$

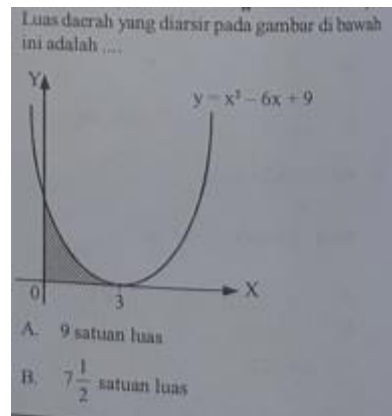
(b)

Gambar 4.2 (a) Matriks RGB sebelum praproses (b) Matriks RGB setelah praproses

1.2 Deskripsi Data Hasil Preprocessing

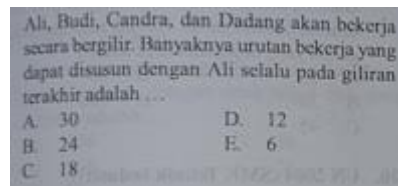
Data yang didapat dari *preprocessing* adalah data hasil *feature extraction* berwarna hitam putih dengan ukuran pixel 500x500. Nilai *feature* ekstraksi yang semakin besar menunjukkan bahwa warna pada gambar yang dilakukan fitur ekstraksi semakin putih. Nilai fitur ekstraksi yang semakin kecil menunjukkan bahwa warna pada gambar yang dilakukan fitur ekstraksi semakin hitam. Label yang digunakan sebanyak 2 label, yaitu label bergambar dan label tidak

bergambar. Jumlah label yang digunakan sebanyak 80 label integral dan 80 label peluang. Berikut merupakan contoh gambar soal integral.



Gambar 4.2 contoh soal integral

Gambar 4.2 menunjukkan bahwa gambar soal integral memiliki beberapa karakteristik. Soal integral memiliki gambar seperti yang ditunjukkan pada kurva garis, memiliki persamaan, memiliki simbol dan teks. Karakteristik tersebut akan digunakan untuk menunjukkan kekhasan gambar soal integral saat membentuk model. Berikut merupakan contoh soal peluang.

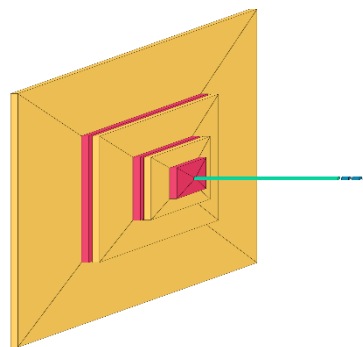


Gambar 4.3 Contoh soal peluang

Gambar 4.3 menunjukkan contoh soal peluang. Karakteristik soal peluang hanya memiliki teks yang disusun oleh huruf dan angka saja. Karakteristik tersebut akan digunakan untuk menjadi ciri khas gambar soal peluang dalam pemodelan.

1.3 Klasifikasi Label Gambar Soal Menggunakan CNN

Setelah melakukan preprocessing data pada setiap *feature* yang digunakan, akan dilakukan pengklasifikasian label soal integral dan peluang dengan menggunakan metode *Convolutional Neural Network* (CNN). Jumlah *convolution kernel* yang digunakan pada tahap ini sejumlah 2,3,4,5,6, dan 7 *convolutional kernel size* dengan masing-masing *convolutional layer* maksimal 3 *convolutional layer*. Berikut merupakan contoh arsitektur CNN yang digunakan dalam penelitian ini.



Gambar 4.4 Contoh arsitektur CNN

Gambar 4.4 menunjukkan contoh visualisasi arsitektur CNN yang digunakan dalam penelitian ini. Arsitektur yang dijadikan contoh pada gambar 4.4 adalah arsitektur dengan 2 *Convolutional kernel* dan 3 *convolutional layer*. Berikut merupakan contoh perhitungan matriks konvolusi pada penelitian ini.

$$\begin{pmatrix} 254 & 251 & \dots & 253 \\ \vdots & \vdots & \ddots & \vdots \\ 253 & 253 & \dots & 253 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \quad \begin{pmatrix} 6 & -6 & \dots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

(a) (b) (c)

Gambar 4.5 (a) Matriks konvolusi (b) matriks kernel konvolusi (c) Matriks konvolusi terbaru

Gambar 4.5 (a) merupakan matriks konvolusi yang digunakan sebagai input pada penelitian ini. Matriks kernel konvolusi pada gambar 4.5 (b) merupakan salah satu matriks penciri yang digunakan dalam penelitian ini. Matriks konvolusi akan dikalikan dengan matriks kernel konvolusi sehingga menghasilkan matriks konvolusi terbaru pada gambar 4.5 (c). berikut merupakan gambaran pada proses matriks ReLU.

$$\begin{pmatrix} 6 & -6 & \dots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix} \quad \begin{pmatrix} 6 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

(a) (b)

Gambar 4.6 (a) matriks konvolusi (b) matriks ReLU

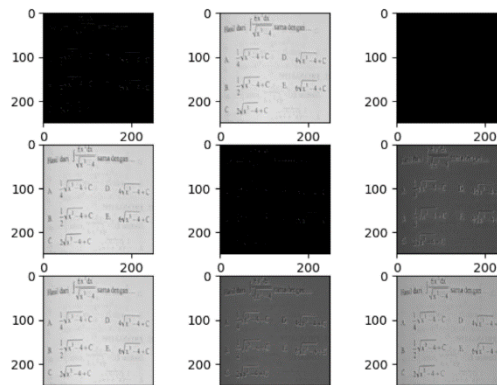
Gambar 4.6 merupakan gambar yang menunjukkan proses perubahan dari matriks konvolusi menjadi matriks ReLU. Pada gambar 4.6 (a) masih terdapat nilai negative seperti -6 dan -1. Gambar 4.6 (b) menunjukkan bahwa matriks konvolusi sudah diubah menjadi matriks ReLU karena sudah tidak memiliki nilai negatif. Berikut merupakan ilustrasi dari pengubahan matriks ReLU menjadi matriks *pooling* menggunakan *max pooling*.

$$\begin{pmatrix} 6 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} \quad \begin{pmatrix} 6 & 0 & 5 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}$$

(a) (b)

Gambar 4.7 (a) matriks ReLU (b) matriks konvolusi terbaru

Gambar 4.7 (a) menunjukkan matriks ReLU yang didapatkan dari proses ReLU. Gambar 4.7 (b) menunjukkan bahwa matriks ReLU sudah diubah menjadi matriks konvolusi terbaru menggunakan metode max pooling. Matriks tersebut akan digunakan pada tahapan selanjutnya dalam proses *neural network*. Berikut merupakan bentuk visualisasi pencirian yang terjadi pada gambar yang digunakan pada penelitian ini.



Gambar 4.8 Proses pencirian pada salah gambar

Berikut pembahasan lebih lanjut untuk masing-masing model yang dibentuk dari arsitektur.

4.3.1 Klasifikasi Label Gambar Soal menggunakan CNN dengan 2 *convolutional kernel*

Berikut merupakan tabel hasil klasifikasi model CNN dengan 2 *convolutional kernel* pada arsitektur CNN dengan maksimal 3 *convolutional layer*.

Tabel 4.1 Hasil kebaikan klasifikasi pada data *training* dan *testing* menggunakan 2 *convolutional kernel*

| No. | <i>Convolutional Layer</i> | Akurasi | |
|-----|----------------------------|-----------------|----------------|
| | | <i>Training</i> | <i>Testing</i> |
| 1. | 1 | 0,5000 | 0,5000 |
| 2. | 2 | 0,9375 | 0,8125 |
| 3. | 3 | 0,9453 | 0,9375 |

Tabel 4.1 menunjukkan hasil kebaikan klasifikasi model CNN dengan 2 *convolutional kernel* pada arsitektur CNN dengan maksimal 3 *convolutional layer*. Pada model-model tersebut, didapatkan bahwa model CNN dengan 2 *convolutional kernel* dan 3 *convolutional layer* merupakan model terbaik dengan nilai akurasi pada data *training* dan *testing* masing-masing sebesar 0,9453 dan 0,9375.

4.3.2 Klasifikasi Label Gambar Soal menggunakan CNN dengan 3 *convolutional kernel*

Berikut merupakan tabel hasil klasifikasi model CNN dengan 3 *convolutional kernel* pada arsitektur CNN dengan maksimal 3 *convolutional layer*.

Tabel 4.2 Hasil kebaikan klasifikasi pada data *training* dan *testing* menggunakan 3 *convolutional kernel*

| No. | <i>Convolutional Layer</i> | Akurasi | |
|-----|----------------------------|-----------------|----------------|
| | | <i>Training</i> | <i>Testing</i> |
| 1. | 1 | 0,7188 | 0,5000 |
| 2. | 2 | 0,5156 | 0,7500 |
| 3. | 3 | 0,8750 | 0,6875 |

Tabel 4.2 menunjukkan bahwa nilai hasil kebaikan klasifikasi pada model CNN dengan 3 *convolutional kernel* pada arsitektur dengan maksimal 3 *convolutional layer* mendapatkan nilai akurasi pada data *training* dan *testing* masing-masing sebesar 0,7188 dan 0,5000 pada arsitektur dengan 1 *convolutional layer*. Model CNN 3 *convolutional kernel* dengan 2 *convolutional layer* mendapatkan hasil kebaikan klasifikasi sebesar 0,5156 pada data *training* dan 0,7500 pada data *testing*. Sedangkan model CNN 3 *convolutional kernel* dengan 3 *convolutional layer* mendapatkan nilai hasil kebaikan klasifikasi sebesar 0,8750 pada data *training* dan 0,6875 pada data *testing*.

4.3.3 Klasifikasi Label Gambar Soal menggunakan CNN dengan 4 *convolutional kernel*

Berikut merupakan tabel hasil klasifikasi model CNN pada data label soal integral dan peluang dengan 4 *convolutional kernel* pada arsitektur maksimal 3 *convolutional layer*.

Tabel 4.3 Hasil kebaikan klasifikasi pada data *training* dan *testing* menggunakan 4 *convolutional kernel*

| No. | <i>Convolutional Layer</i> | Akurasi | |
|-----|----------------------------|-----------------|----------------|
| | | <i>Training</i> | <i>Testing</i> |
| 1. | 1 | 0,6172 | 0,7812 |
| 2. | 2 | 0,6406 | 0,5000 |
| 3. | 3 | 0,5391 | 0,5000 |

Tabel 4.3 menunjukkan nilai hasil kebaikan klasifikasi model CNN dengan 4 *convolutional kernel* pada arsitektur maksimal 3 *convolutional layer*. Model CNN 4 *convolutional kernel* dengan 1 *convolutional layer* mendapatkan hasil kebaikan klasifikasi pada data *training* dan *testing* masing-masing sebesar 0,6172 dan 0,7812. Model CNN dengan 4 *convolutional kernel* dengan 2 *convolutional layer* mendapatkan hasil kebaikan klasifikasi pada data *training* dan data *testing* masing-masing sebesar 0,6406 dan 0,5000. Model CNN dengan 4 *convolutional kernel* dengan 3 *convolutional layer* mendapatkan hasil klasifikasi pada data *training* dan *testing* masing-masing sebesar 0,5391 dan 0,5000. Model CNN terbaik pada data *training* didapatkan oleh model CNN 4 *convolutional kernel* dengan 2 *convolutional layer* dan model CNN terbaik pada data *testing* didapatkan oleh model CNN dengan 4 *convolutional kernel* dengan 1 *convolutional layer*.

4.3.4 Klasifikasi Gambar Soal menggunakan CNN dengan 5 *convolutional kernel*

Berikut merupakan hasil kebaikan klasifikasi model CNN dengan 5 *convolutional kernel* dengan maksimal 3 *convolutional layer* pada data berlabel integral dan peluang.

Tabel 4.4 Hasil kebaikan klasifikasi pada data *training* dan *testing* menggunakan 5 *convolutional kernel*

| No. | <i>Convolutional Layer</i> | Akurasi | |
|-----|----------------------------|-----------------|----------------|
| | | <i>Training</i> | <i>Testing</i> |
| 1. | 1 | 0,7031 | 0,5312 |
| 2. | 2 | 0,5000 | 0,5000 |
| 3. | 3 | 0,8047 | 0,5000 |

Tabel 4.4 menunjukkan hasil kebaikan klasifikasi pada data *training* dan data *testing* label soal integral dan peluang menggunakan 5 *convolutional kernel* dengan maksimal 3 *convolutional layer*. Model CNN terbaik yang didapatkan pada pengklasifikasian data *training* adalah model CNN menggunakan 5 *convolutional kernel* dengan 3 *convolutional layer* yang mendapatkan nilai akurasi 0,8047. Model CNN terbaik yang mengklasifikasi data *testing* adalah model CNN menggunakan 5 *convolutional kernel* dengan 1 *convolutional layer*.

4.3.5 Klasifikasi Gambar Soal menggunakan CNN dengan 6 *convolutional kernel*

Berikut merupakan tabel nilai hasil kebaikan klasifikasi model CNN menggunakan 6 *convolutional kernel* dengan maksimal menggunakan 3 *convolutional layer* pada data label soal integral dan peluang.

Tabel 4.5 Hasil kebaikan klasifikasi pada data *training* dan *testing* menggunakan 6 *convolutional kernel*

| No. | <i>Convolutional Layer</i> | Akurasi | |
|-----|----------------------------|-----------------|----------------|
| | | <i>Training</i> | <i>Testing</i> |
| 1. | 1 | 0,6406 | 0,7500 |
| 2. | 2 | 0,5000 | 0,5000 |
| 3. | 3 | 0,8984 | 0,8125 |

Tabel 4.5 menunjukkan nilai hasil kebaikan klasifikasi pada data *training* dan *testing* menggunakan 6 *convolutional kernel* dengan maksimal 3 *convolutional layer*. Model terbaik pada pengklasifikasian data *training* dan data *testing* didapatkan oleh model CNN dengan 6 *convolutional kernel* dengan 3 *convolutional layer* dengan nilai akurasi sebesar 0,8984 untuk data *training* dan 0,8125 untuk data *testing*.

4.3.6 Klasifikasi Gambar Soal menggunakan CNN dengan 7 *convolutional kernel*

Berikut merupakan tabel hasil kebaikan klasifikasi model CNN dengan 7 *convolutional kernel* dengan maksimal 3 *convolutional layer* pada data label soal integral peluang.

Tabel 4.6 Hasil kebaikan klasifikasi pada data *training* dan *testing* menggunakan 7 *convolutional kernel*

| No. | <i>Convolutional Layer</i> | Akurasi | |
|-----|----------------------------|-----------------|----------------|
| | | <i>Training</i> | <i>Testing</i> |
| 1. | 1 | 0,5000 | 0,5000 |
| 2. | 2 | 0,5781 | 0,8750 |
| 3. | 3 | 0,5000 | 0,5625 |

Tabel 4.6 menunjukkan bahwa nilai hasil kebaikan klasifikasi model CNN dengan 7 *convolutional kernel* terbaik didapatkan oleh model dengan arsitektur 2 *convolutional layer* yang mendapatkan nilai kebaikan pada data *training* sebesar 0,5781 dan nilai kebaikan hasil klasifikasi pada data *testing* sebesar 0,8750.

1.4 Perbandingan Performa Klasifikasi Setiap *Convolutional Kernel*

Setelah dilakukan pengklasifikasian menggunakan model *Convolutional Neural Network* dengan jumlah *convolution kernel* sejumlah 2,3,4,5,6 dan 7 dalam jumlah *convolutional layer* sejumlah maksimal 3 dan didapatkan nilai kebaikan klasifikasi berupa akurasi pada masing-masing model CNN maka akan dilakukan perbandingan model untuk mendapatkan model terbaik. Berikut merupakan tabel perbandingan performansi model CNN.

Tabel 4.7 Perbandingan performansi model CNN

| No. | <i>Convolutional Kernel</i> | <i>Convolutional layer</i> | Akurasi | |
|-----|-----------------------------|----------------------------|-----------------|----------------|
| | | | <i>Training</i> | <i>Testing</i> |
| 1. | 2 | 3 | 0,9453 | 0,9375 |
| 2. | 3 | 3 | 0,8750 | 0,6875 |
| 3. | 4 | 1 | 0,6172 | 0,7812 |
| 4. | 5 | 3 | 0,8047 | 0,5000 |
| 5. | 6 | 3 | 0,8984 | 0,8125 |
| 6. | 7 | 2 | 0,5781 | 0,8750 |

Tabel 4.7 menunjukkan bahwa model *convolutional neural network* dengan *convolution kernel* dan *convolutional layer* masing-masing sebanyak 2 dan 3 merupakan model terbaik pada data *training* dan *testing* dengan nilai akurasi masing-masing 0,9453 dan 0,9375.