**Nama : Shabrina Katresnawati**

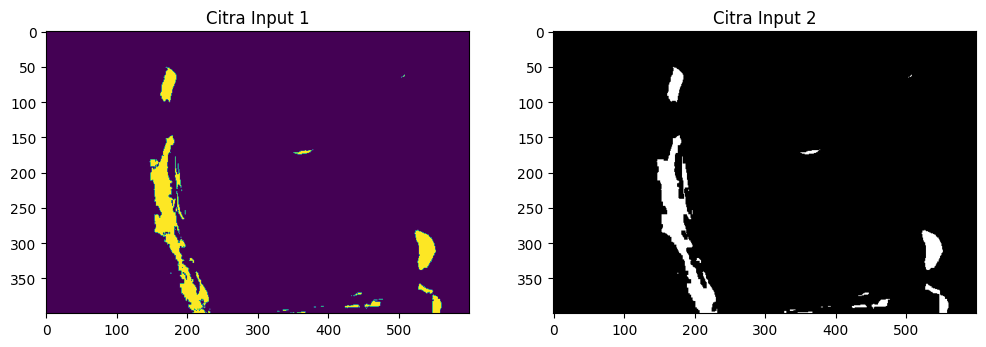
**NIM : 1207070136**

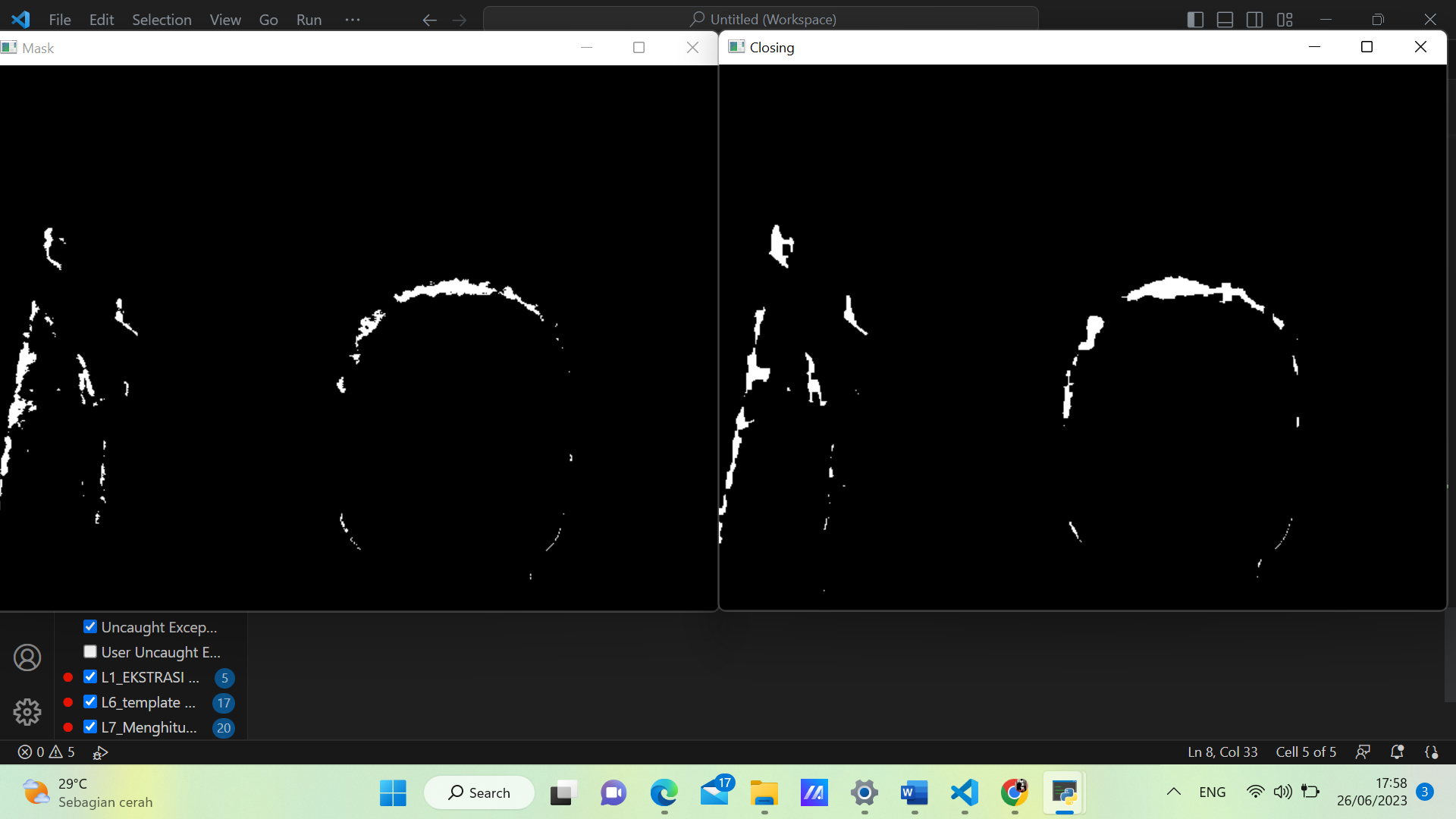
**Kelas : TKK**

**TUGAS 11 PRAKTIKUM PENGOLAHAN CITRA DIGITAL**

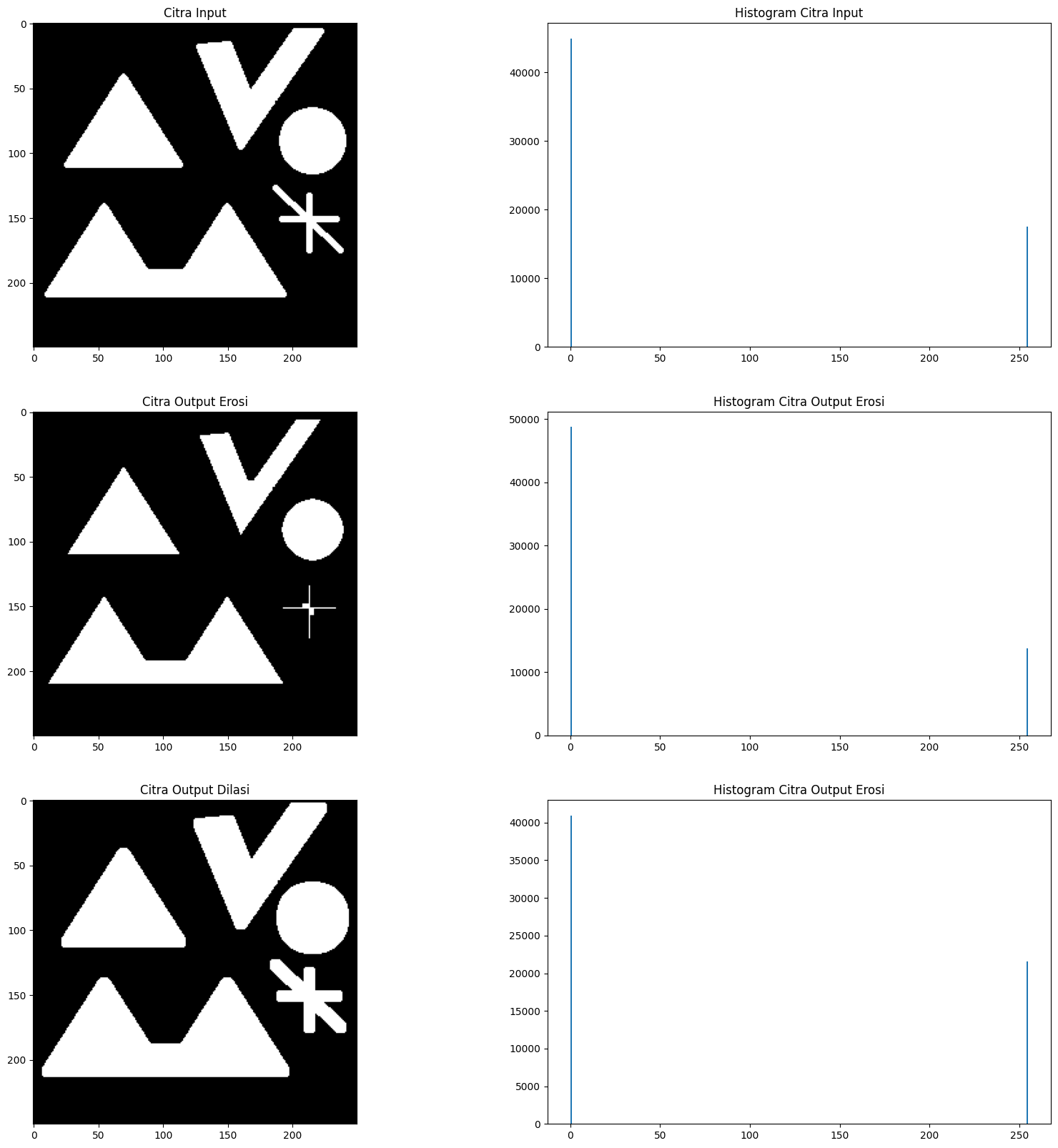
1. **Percobaan Morfologi Closing**

****

****

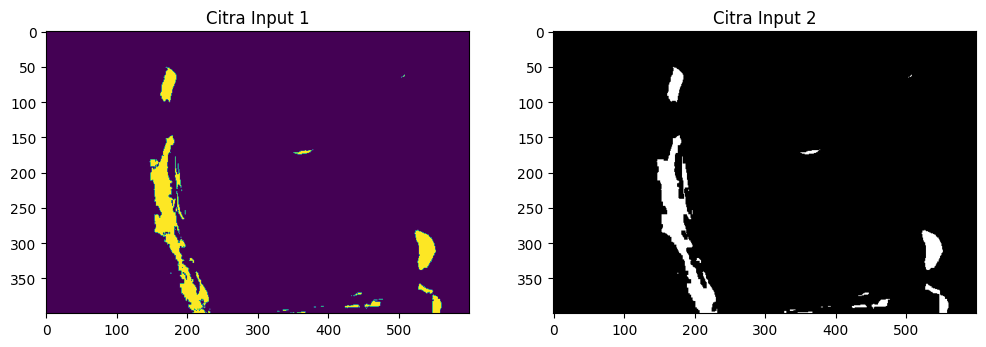
****

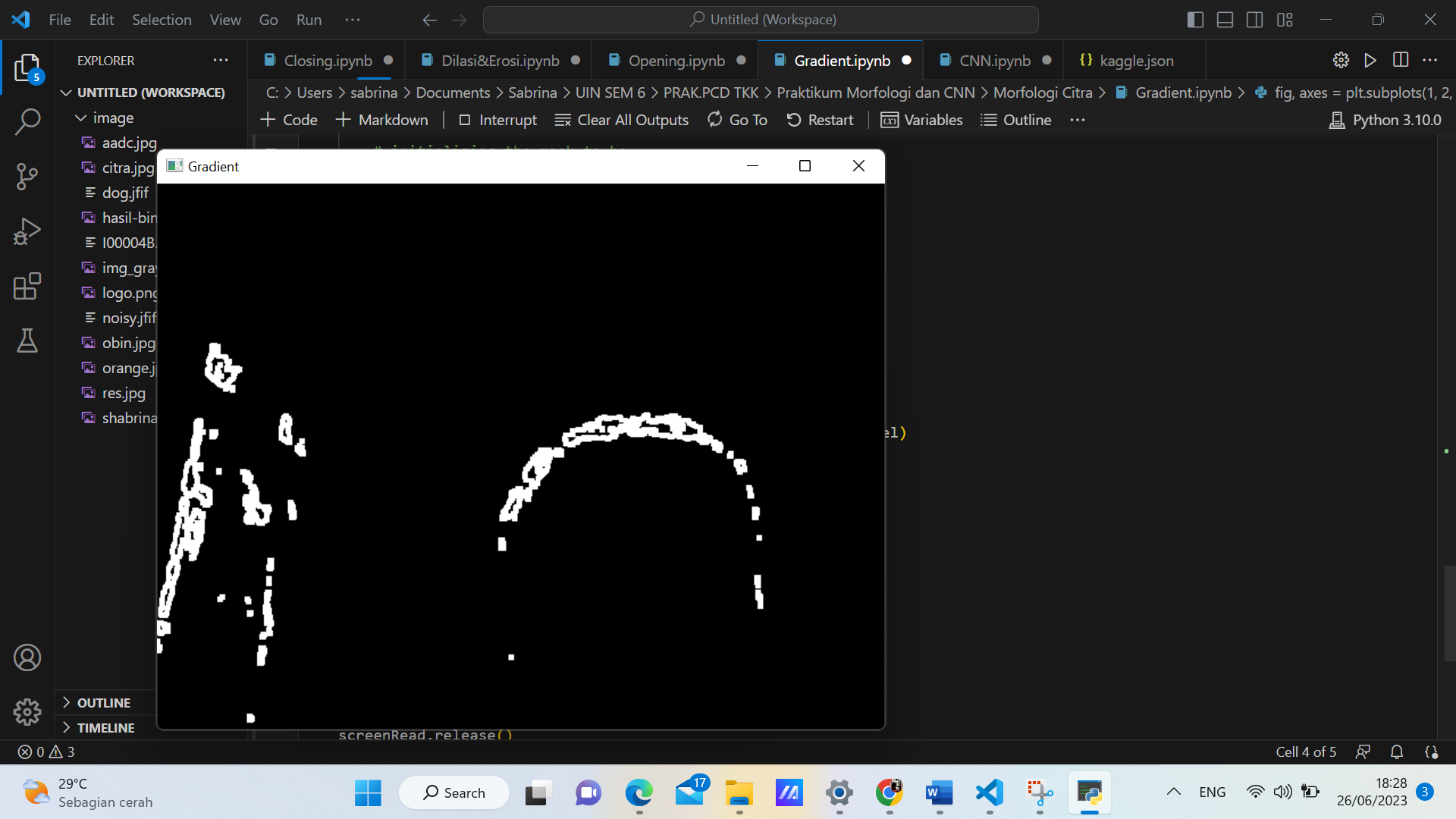
1. **Morfologi Dilasi & Erosi**

****

1. **Morfologi Gradient**

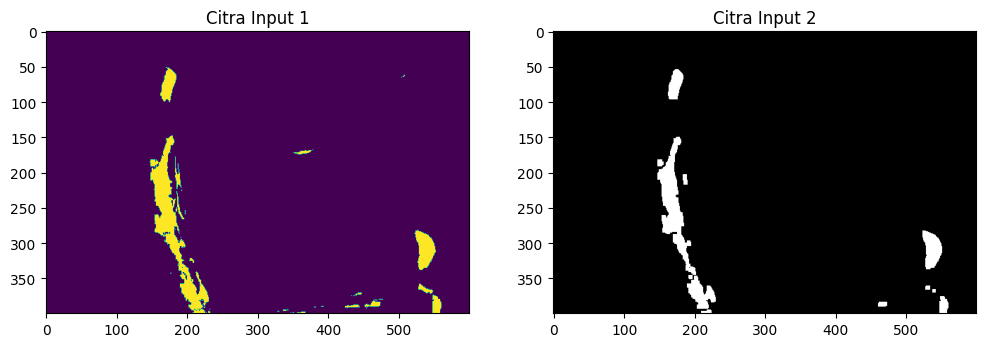
****

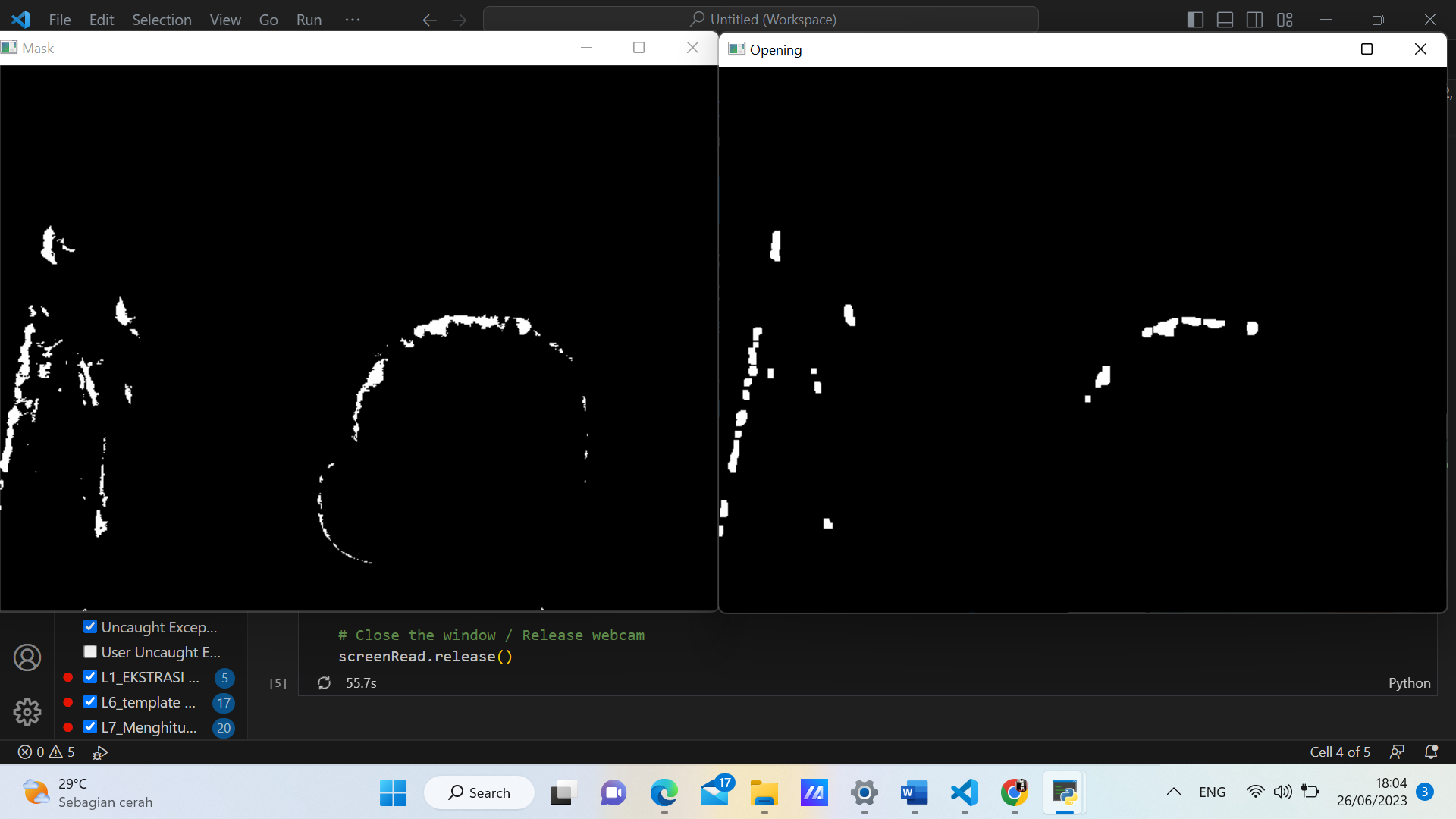
****

****

1. **Morfologi Opening**

****

****



**Analisis :**

Dari keempat metode yang telah dicoba, yaitu Gradient Morphological Operation, Closing Morphological Operation, Erosion and Dilation, dan Opening Morphological Operation, masing-masing metode memiliki efek dan tujuan yang berbeda dalam memproses gambar.

1. Gradient Morphological Operation:

Metode ini digunakan untuk menyoroti perbedaan intensitas di sekitar tepi objek pada gambar. Menggunakan operasi morfologi dilasi dan erosi, metode ini menghasilkan citra yang menunjukkan perbedaan intensitas di sekitar tepi objek. Hasilnya adalah citra yang menyoroti kontur dan garis tepi dari objek dalam gambar.

2. Closing Morphological Operation:

Metode ini digunakan untuk menghilangkan lubang kecil dan menghubungkan bagian-bagian yang terpisah dalam objek pada gambar. Menggunakan operasi morfologi dilasi diikuti oleh erosi, metode ini menghasilkan citra yang lebih halus dengan mengisi celah-celah kecil dan menghubungkan bagian-bagian yang terputus dalam objek.

3. Erosion and Dilation:

Metode ini adalah kombinasi dari dua operasi morfologi dasar, yaitu erosi dan dilasi. Erosi digunakan untuk menghilangkan piksel-piksel yang memiliki intensitas rendah dan menghasilkan pengecilan objek, sedangkan dilasi digunakan untuk mengisi celah-celah dan mengembalikan ukuran objek. Kombinasi kedua operasi ini dapat mengubah bentuk, ukuran, dan struktur objek dalam gambar.

4. Opening Morphological Operation:

Metode ini adalah kombinasi dari erosi dan dilasi dalam urutan yang berbeda. Erosi digunakan terlebih dahulu untuk menghilangkan noise dan menghaluskan objek, dan dilasi digunakan untuk mengembalikan ukuran dan bentuk objek yang hilang selama erosi. Metode ini berguna untuk membersihkan objek dari noise dan menghaluskan struktur objek dalam gambar.

Analisis perbandingan hasil gambar dari keempat metode tersebut tergantung pada konteks dan tujuan pengolahan gambar. Setiap metode memiliki efek yang unik dan dapat digunakan untuk mencapai hasil yang diinginkan. Gradient Morphological Operation menyoroti tepi objek, Closing Morphological Operation menghilangkan lubang dan menghubungkan objek, Erosion and Dilation mengubah ukuran dan struktur objek, dan Opening Morphological Operation membersihkan objek dari noise dan menghaluskan struktur.

Berdasarkan percobaan di atas, dapat disimpulkan bahwa metode Clossing memiliki hasil terbaik karena menggunakan bintik pendeteksi kecil yang memungkinkan akurasi yang lebih tinggi, meskipun pada percobaan ini hasilnya tidak optimal. Metode Dilasi dan Erosi tidak dapat dibandingkan dengan ketiga metode lainnya karena menggunakan objek yang berbeda dalam percobaan ini. Metode Gradient awalnya tampak memiliki hasil terbaik, tetapi memiliki bintik pendeteksi yang lebih besar dibandingkan dengan metode lainnya, sehingga tidak dapat mendeteksi dengan presisi maksimal seperti metode lainnya. Terakhir, metode Opening memiliki kesamaan dengan metode Clossing, menggunakan bintik pendeteksi kecil yang menghasilkan akurasi yang lebih tinggi dibandingkan dengan metode gradient.

**Convolution Neural Network (CNN)**

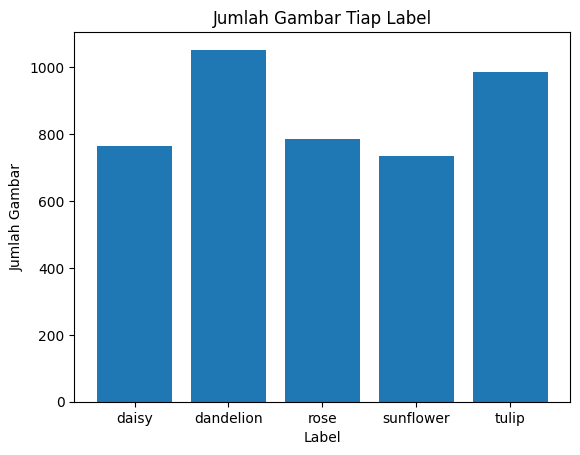
**Klasifikasi Jenis Bunga**

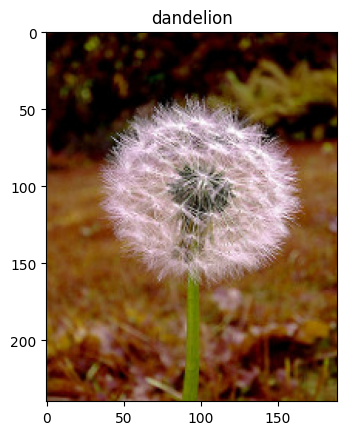
1. **Data Understanding**

flowers.zip: Skipping, found more recently modified local copy (use --force to force download)

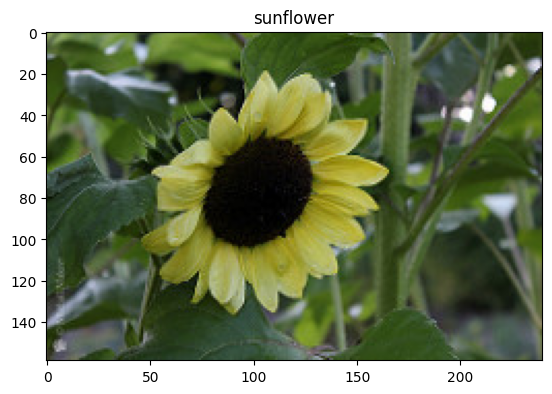
['dandelion', 'tulip', 'sunflower', 'daisy', 'rose']

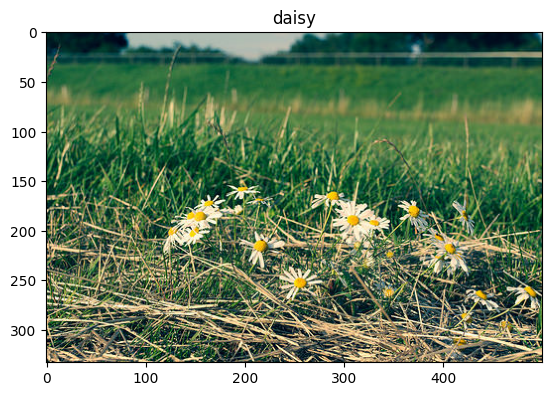
Total Files : 4317

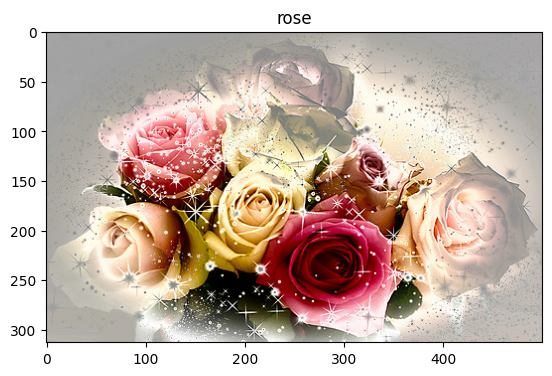












1. **Data Preparation**

Found 3457 images belonging to 5 classes.

Found 860 images belonging to 5 classes.

1. **Modeling**

Epoch 1/20

109/109 [==============================] - 36s 166ms/step - loss: 1.3845 - accuracy: 0.3813 - val\_loss: 1.2844 - val\_accuracy: 0.4512

Epoch 2/20

109/109 [==============================] - 17s 156ms/step - loss: 1.1337 - accuracy: 0.5276 - val\_loss: 1.0611 - val\_accuracy: 0.5628

Epoch 3/20

109/109 [==============================] - 19s 174ms/step - loss: 1.0161 - accuracy: 0.5962 - val\_loss: 1.0647 - val\_accuracy: 0.5686

Epoch 4/20

109/109 [==============================] - 17s 153ms/step - loss: 0.9334 - accuracy: 0.6407 - val\_loss: 1.0283 - val\_accuracy: 0.6128

Epoch 5/20

109/109 [==============================] - 20s 184ms/step - loss: 0.8749 - accuracy: 0.6581 - val\_loss: 0.9742 - val\_accuracy: 0.6233

Epoch 6/20

109/109 [==============================] - 17s 154ms/step - loss: 0.8937 - accuracy: 0.6488 - val\_loss: 0.8708 - val\_accuracy: 0.6581

Epoch 7/20

109/109 [==============================] - 17s 158ms/step - loss: 0.7992 - accuracy: 0.6893 - val\_loss: 0.8396 - val\_accuracy: 0.6930

Epoch 8/20

109/109 [==============================] - 16s 151ms/step - loss: 0.8246 - accuracy: 0.6772 - val\_loss: 0.8394 - val\_accuracy: 0.6837

Epoch 9/20

109/109 [==============================] - 17s 154ms/step - loss: 0.7689 - accuracy: 0.7067 - val\_loss: 0.8158 - val\_accuracy: 0.6953

Epoch 10/20

109/109 [==============================] - 17s 152ms/step - loss: 0.7116 - accuracy: 0.7243 - val\_loss: 0.7713 - val\_accuracy: 0.7081

Epoch 11/20

109/109 [==============================] - 17s 153ms/step - loss: 0.6670 - accuracy: 0.7431 - val\_loss: 0.8402 - val\_accuracy: 0.6953

Epoch 12/20

109/109 [==============================] - 17s 160ms/step - loss: 0.6572 - accuracy: 0.7454 - val\_loss: 0.8017 - val\_accuracy: 0.7105

Epoch 13/20

109/109 [==============================] - 17s 153ms/step - loss: 0.6264 - accuracy: 0.7590 - val\_loss: 0.7493 - val\_accuracy: 0.7256

Epoch 14/20

109/109 [==============================] - 16s 151ms/step - loss: 0.6173 - accuracy: 0.7616 - val\_loss: 0.7643 - val\_accuracy: 0.7070

Epoch 15/20

109/109 [==============================] - 17s 160ms/step - loss: 0.5762 - accuracy: 0.7706 - val\_loss: 0.7590 - val\_accuracy: 0.7256

Epoch 16/20

109/109 [==============================] - 17s 154ms/step - loss: 0.5879 - accuracy: 0.7651 - val\_loss: 0.8011 - val\_accuracy: 0.7151

Epoch 17/20

109/109 [==============================] - 17s 159ms/step - loss: 0.5564 - accuracy: 0.7825 - val\_loss: 0.8921 - val\_accuracy: 0.7209

Epoch 18/20

109/109 [==============================] - 17s 151ms/step - loss: 0.5408 - accuracy: 0.7984 - val\_loss: 0.7858 - val\_accuracy: 0.7198

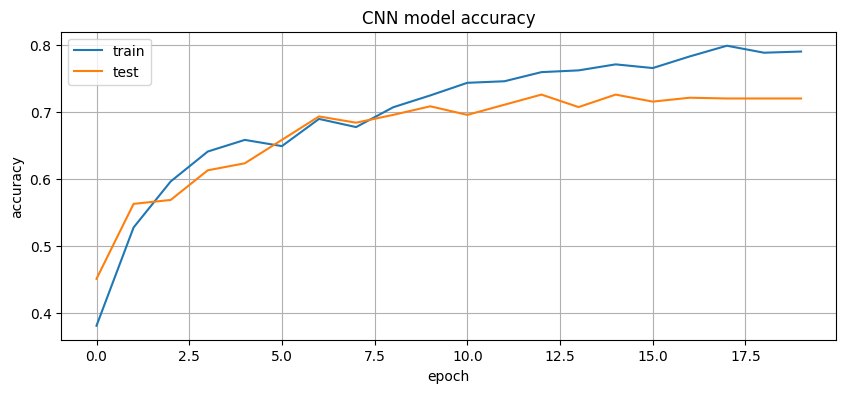
Epoch 19/20

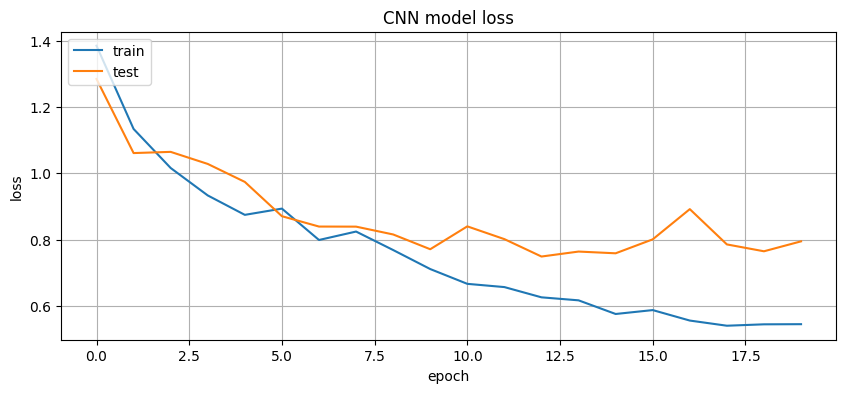
109/109 [==============================] - 17s 156ms/step - loss: 0.5450 - accuracy: 0.7880 - val\_loss: 0.7651 - val\_accuracy: 0.7198

Epoch 20/20

109/109 [==============================] - 19s 174ms/step - loss: 0.5454 - accuracy: 0.7897 - val\_loss: 0.7951 - val\_accuracy: 0.7198

1. **Evaluasi Model**





poch 1/10

109/109 [==============================] - 31s 229ms/step - loss: 1.0848 - accuracy: 0.5722 - val\_loss: 0.7033 - val\_accuracy: 0.7337

Epoch 2/10

109/109 [==============================] - 26s 234ms/step - loss: 0.7828 - accuracy: 0.7104 - val\_loss: 0.6269 - val\_accuracy: 0.7686

Epoch 3/10

109/109 [==============================] - 21s 189ms/step - loss: 0.6781 - accuracy: 0.7495 - val\_loss: 0.6042 - val\_accuracy: 0.7756

Epoch 4/10

109/109 [==============================] - 20s 187ms/step - loss: 0.6556 - accuracy: 0.7596 - val\_loss: 0.6269 - val\_accuracy: 0.7605

Epoch 5/10

109/109 [==============================] - 22s 197ms/step - loss: 0.6536 - accuracy: 0.7561 - val\_loss: 0.5528 - val\_accuracy: 0.8023

Epoch 6/10

109/109 [==============================] - 21s 188ms/step - loss: 0.6065 - accuracy: 0.7744 - val\_loss: 0.5643 - val\_accuracy: 0.7837

Epoch 7/10

109/109 [==============================] - 21s 190ms/step - loss: 0.5891 - accuracy: 0.7868 - val\_loss: 0.5939 - val\_accuracy: 0.7814

Epoch 8/10

109/109 [==============================] - 20s 183ms/step - loss: 0.5996 - accuracy: 0.7807 - val\_loss: 0.5423 - val\_accuracy: 0.7953

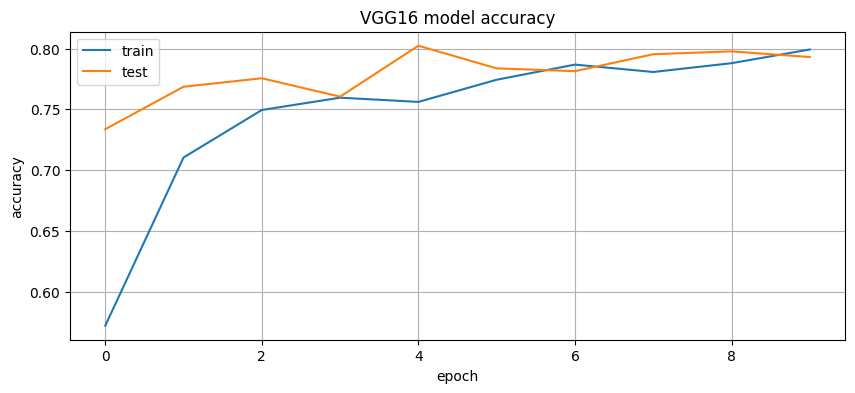
Epoch 9/10

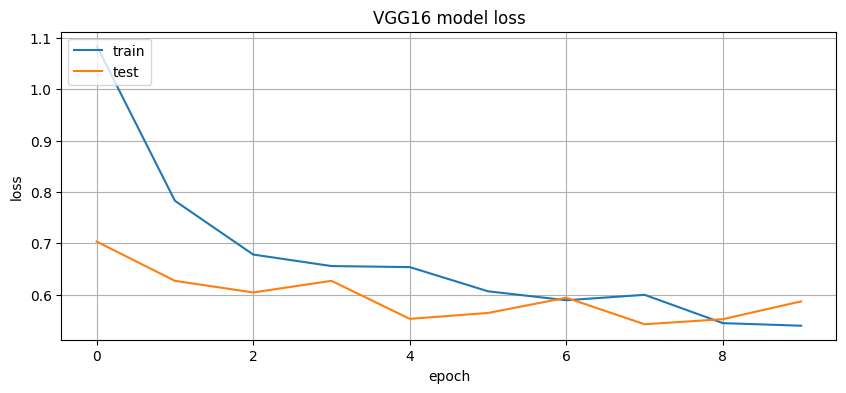
109/109 [==============================] - 21s 192ms/step - loss: 0.5445 - accuracy: 0.7880 - val\_loss: 0.5522 - val\_accuracy: 0.7977

Epoch 10/10

109/109 [==============================] - 22s 200ms/step - loss: 0.5394 - accuracy: 0.7992 - val\_loss: 0.5869 - val\_accuracy: 0.7930

1. **Transfer Learning Menggunakan VGG16**





Epoch 1/10

109/109 [==============================] - 28s 203ms/step - loss: 1.6895 - accuracy: 0.2705 - val\_loss: 1.5449 - val\_accuracy: 0.2570

Epoch 2/10

109/109 [==============================] - 20s 184ms/step - loss: 1.5668 - accuracy: 0.3092 - val\_loss: 1.4682 - val\_accuracy: 0.3535

Epoch 3/10

109/109 [==============================] - 18s 169ms/step - loss: 1.5556 - accuracy: 0.3422 - val\_loss: 1.4853 - val\_accuracy: 0.3267

Epoch 4/10

109/109 [==============================] - 19s 174ms/step - loss: 1.5072 - accuracy: 0.3465 - val\_loss: 1.4679 - val\_accuracy: 0.3826

Epoch 5/10

109/109 [==============================] - 18s 169ms/step - loss: 1.5091 - accuracy: 0.3549 - val\_loss: 1.5069 - val\_accuracy: 0.3872

Epoch 6/10

109/109 [==============================] - 19s 174ms/step - loss: 1.5156 - accuracy: 0.3413 - val\_loss: 1.4387 - val\_accuracy: 0.3860

Epoch 7/10

109/109 [==============================] - 18s 168ms/step - loss: 1.4804 - accuracy: 0.3535 - val\_loss: 1.4330 - val\_accuracy: 0.4047

Epoch 8/10

109/109 [==============================] - 19s 175ms/step - loss: 1.4463 - accuracy: 0.3787 - val\_loss: 1.5655 - val\_accuracy: 0.3895

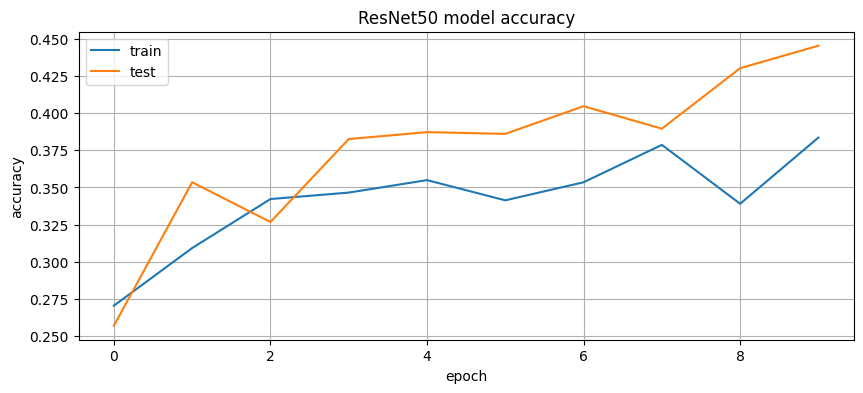
Epoch 9/10

109/109 [==============================] - 18s 169ms/step - loss: 1.5083 - accuracy: 0.3390 - val\_loss: 1.4432 - val\_accuracy: 0.4302

Epoch 10/10

109/109 [==============================] - 19s 175ms/step - loss: 1.4408 - accuracy: 0.3836 - val\_loss: 1.3676 - val\_accuracy: 0.4453

1. **Transfer Learning Menggunakan ResNet50**





Epoch 1/20

109/109 [==============================] - 44s 260ms/step - loss: 0.8716 - accuracy: 0.6653 - val\_loss: 0.4699 - val\_accuracy: 0.8326

Epoch 2/20

109/109 [==============================] - 20s 186ms/step - loss: 0.5478 - accuracy: 0.7998 - val\_loss: 0.3687 - val\_accuracy: 0.8756

Epoch 3/20

109/109 [==============================] - 20s 179ms/step - loss: 0.4326 - accuracy: 0.8354 - val\_loss: 0.4137 - val\_accuracy: 0.8628

Epoch 4/20

109/109 [==============================] - 20s 180ms/step - loss: 0.3868 - accuracy: 0.8577 - val\_loss: 0.4668 - val\_accuracy: 0.8500

Epoch 5/20

109/109 [==============================] - 20s 186ms/step - loss: 0.3669 - accuracy: 0.8652 - val\_loss: 0.3780 - val\_accuracy: 0.8709

Epoch 6/20

109/109 [==============================] - 20s 179ms/step - loss: 0.3448 - accuracy: 0.8701 - val\_loss: 0.3630 - val\_accuracy: 0.8674

Epoch 7/20

109/109 [==============================] - 20s 184ms/step - loss: 0.3638 - accuracy: 0.8620 - val\_loss: 0.4531 - val\_accuracy: 0.8558

Epoch 8/20

109/109 [==============================] - 19s 178ms/step - loss: 0.3437 - accuracy: 0.8750 - val\_loss: 0.3551 - val\_accuracy: 0.8802

Epoch 9/20

109/109 [==============================] - 20s 184ms/step - loss: 0.2951 - accuracy: 0.8889 - val\_loss: 0.3933 - val\_accuracy: 0.8791

Epoch 10/20

109/109 [==============================] - 21s 192ms/step - loss: 0.3088 - accuracy: 0.8878 - val\_loss: 0.3521 - val\_accuracy: 0.8872

Epoch 11/20

109/109 [==============================] - 20s 179ms/step - loss: 0.2937 - accuracy: 0.8941 - val\_loss: 0.3648 - val\_accuracy: 0.8872

Epoch 12/20

109/109 [==============================] - 20s 184ms/step - loss: 0.3073 - accuracy: 0.8837 - val\_loss: 0.3592 - val\_accuracy: 0.8779

Epoch 13/20

109/109 [==============================] - 19s 178ms/step - loss: 0.3120 - accuracy: 0.8840 - val\_loss: 0.3631 - val\_accuracy: 0.8779

Epoch 14/20

109/109 [==============================] - 20s 186ms/step - loss: 0.2811 - accuracy: 0.8924 - val\_loss: 0.3499 - val\_accuracy: 0.8907

Epoch 15/20

109/109 [==============================] - 21s 190ms/step - loss: 0.2687 - accuracy: 0.9028 - val\_loss: 0.3415 - val\_accuracy: 0.8919

Epoch 16/20

109/109 [==============================] - 20s 179ms/step - loss: 0.2597 - accuracy: 0.9034 - val\_loss: 0.4069 - val\_accuracy: 0.8733

Epoch 17/20

109/109 [==============================] - 20s 185ms/step - loss: 0.2611 - accuracy: 0.9014 - val\_loss: 0.3370 - val\_accuracy: 0.8919

Epoch 18/20

109/109 [==============================] - 19s 177ms/step - loss: 0.2769 - accuracy: 0.8970 - val\_loss: 0.3838 - val\_accuracy: 0.8721

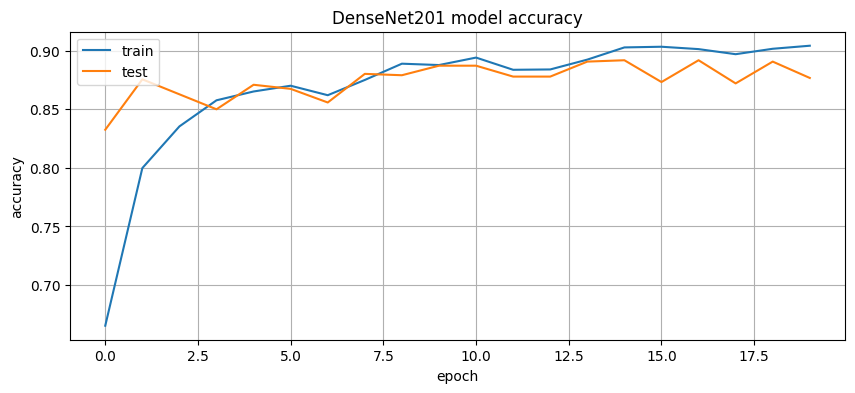
Epoch 19/20

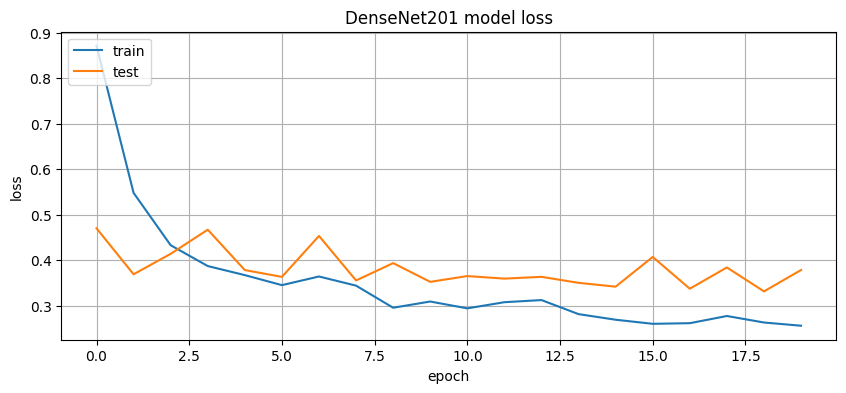
109/109 [==============================] - 20s 186ms/step - loss: 0.2626 - accuracy: 0.9016 - val\_loss: 0.3311 - val\_accuracy: 0.8907

Epoch 20/20

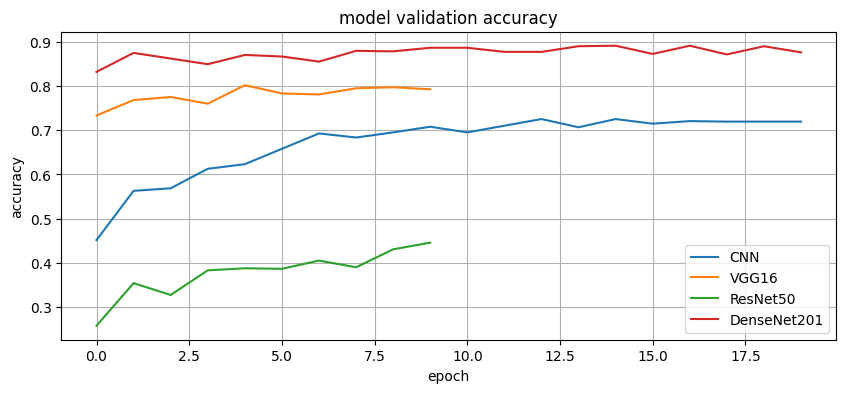
109/109 [==============================] - 20s 184ms/step - loss: 0.2555 - accuracy: 0.9043 - val\_loss: 0.3781 - val\_accuracy: 0.8767

1. **Transfer Learning Menggunakan DenseNet201**

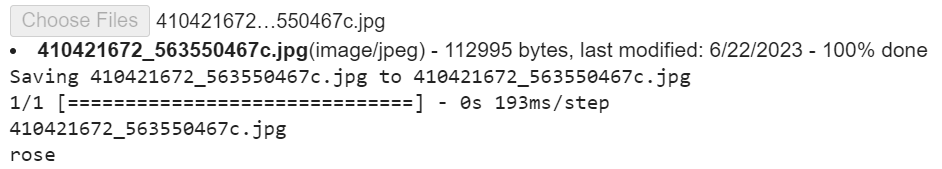


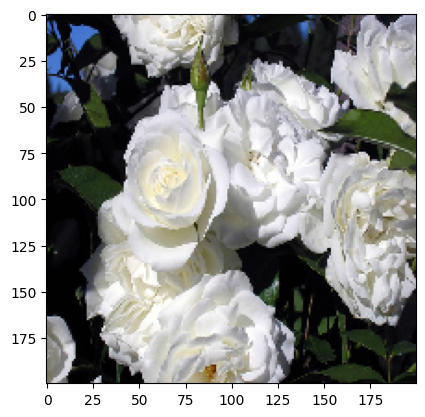


1. **Evaluation**



1. **Uji Coba Model**





Analisis : CNN adalah jenis arsitektur jaringan saraf yang banyak digunakan dalam pengolahan citra digital. CNN dirancang khusus untuk mengatasi masalah pengenalan pola dalam citra, seperti pengenalan objek, segmentasi, klasifikasi, dan deteksi wajah. CNN terinspirasi dari sistem visual manusia dan menggunakan konsep konvolusi untuk ekstraksi fitur penting secara otomatis. Dalam percobaan di atas, proses klasifikasi melibatkan beberapa tahap, mulai dari pemahaman data untuk eksplorasi, ekstraksi fitur, dan segmentasi citra. Tahap Data Preparation digunakan untuk membersihkan, memproses, dan menormalisasi citra, serta ekstraksi fitur yang lebih lanjut. Setelah itu, dilakukan pemodelan dan evaluasi model. Dalam pembuatan arsitektur CNN, terdapat metode Transfer Learning menggunakan VGG16, ResNet50, dan DenseNet201. Metode ini memanfaatkan pengetahuan yang telah dipelajari oleh model tersebut untuk tugas-tugas pengolahan citra baru seperti klasifikasi, deteksi objek, dan lainnya. Metode Transfer Learning memungkinkan adaptasi atau fine-tuning model untuk mencapai performa yang baik dalam pengolahan citra.