

DAFTAR PUSTAKA

- Fansyuri, M., & Yunita, D. (2022). Identifikasi Jenis Buah Apel Berdasarkan Ekstraksi Ciri Bentuk dan Warna Menggunakan Metode Klasifikasi Naive Bayes. *Journal of Artificial Intelligence and Innovative Applications*, 3(3), 186–195. <http://openjournal.unpam.ac.id/index.php/JOAIIA/index>
- H. Abd al karim, M., & A. Karim, A. (2021). Using Texture Feature in Fruit Classification. *Engineering and Technology Journal*, 39(1B), 67–79. <https://doi.org/10.30684/etj.v39i1b.1741>
- Haba, R. K., & Pelangi, K. C. (2020). Pengelompokan Buah Jeruk menggunakan Naïve Bayes dan Gray Level Co-occurrence Matrix. *ILKOM Jurnal Ilmiah*, 12(1), 17–24. <https://doi.org/10.33096/ilkom.v12i1.494.17-24>
- Saputra, A. J., & Andriyani, W. (2025). Fruit Image Classification Using Naive Bayes Algorithm with Histogram of Oriented Gradients (HOG) Feature Extraction. *Journal of Artificial Intelligence and Software Engineering (J-AISE)*, 5(1), 215. <https://doi.org/10.30811/jaise.v5i1.6536>
- (Afriansyah et al. 2023) Afriansyah, M., Joni Saputra, Yuan Sa’adati, and Valian Yoga Pudya Ardhana. 2023. “Optimasi Algoritma Nai?Ve Bayes Untuk Klasifikasi Buah Apel Berdasarkan Fitur Warna RGB.” *Bulletin of Computer Science Research* 3(3):242–49. doi: 10.47065/bulletincsr.v3i3.251.
- (Prajatama et al. 2019) Prajatama, Kholiq, Fajar Eko Nugroho, Andri Faisal Sentosa, and Syifa Fauziah. 2019. “Deteksi Kualitas Buah Apel Malang Manalagi Menggunakan Algoritma Naive Bayes.” *E-Jurnal JUSITI (Jurnal Sistem Informasi Dan Teknologi Informasi)* 8–1(1):32–38. doi: 10.36774/jusiti.v8i1.598.
- (Fatmiatun 2023) Fatmiatun, Husnul. 2023. “Dengan Citra Digital Menggunakan Algoritma Naïve Bayes Ripeness Classification of Banana Level Using Digital Image With Naïve Bayes Algorithm Husnul Fatmiatun Program Studi Informatika Majene Tahun 2023 Abstrak.” 1–27.

(Syuja 2025)Syuja, Muhammad Abid. 2025. “Analisis Perbandingan Akurasi Buah Apel Emas Berdasarkan Kelas Dengan Algoritma Gray Level Co-Occurance (GLCM) Dan Convolutional Neural Network (CNN).” *Inovasi Pendidikan Nusantara* 6(1):523–39.

Li, Y., Feng, X., Liu, Y., & Han, X. (2021). Apple quality identification and classification by image processing based on convolutional neural networks. *Scientific Reports*, 11(1), 1–15. <https://doi.org/10.1038/s41598-021-96103-2>

Hadi, H. P., & Rachmawanto, E. H. (2022). Ekstraksi Fitur Warna Dan Glcm Pada Algoritma Knn Untuk Klasifikasi Kematangan Rambutan. *Jurnal Informatika Polinema*, 8(3), 63–68. <https://doi.org/10.33795/jip.v8i3.949>

LAMPIRAN

```

function gui_klasifikasi_apel

clc; close all;

f = figure('Name', 'Klasifikasi Apel Malang', ...
    'Position', [300 100 1050 600], 'Color', [0.95 0.95 0.95]);

handles.axes1 = axes('Parent', f, 'Units', 'pixels', 'Position', [50 330 200 200]);
title(handles.axes1, 'Citra Asli');

handles.axes2 = axes('Parent', f, 'Units', 'pixels', 'Position', [300 330 200 200]);
title(handles.axes2, 'Citra Grayscale');

handles.uitableHasil = uitable(f, 'Data', {}, 'Position', [50 20 950 280], ...
    'ColumnName', {'No', 'Nama Citra', 'Target', 'Hasil', 'Keterangan', ...
        'Rmean', 'Gmean', 'Bmean', 'Entropy', 'Contrast', 'Correlation', 'Energy', 'Homogen
        eity', 'Smoothness', 'Skewness', 'Intensitas'});

guidata(f, handles);

uicontrol(f, 'Style', 'pushbutton', 'String', 'Training Model', ...
    'Position', [550 480 120 40], 'Callback', @(src, evt) trainingCallback());

uicontrol(f, 'Style', 'pushbutton', 'String', 'Input', ...

```

```

'Position', [550 420 120 40], 'Callback', @(src, evt)
    browseCallback(guidata(f));

uicontrol(f, 'Style', 'pushbutton', 'String', 'Uji Semua', ...
    'Position', [700 420 120 40], 'Callback', @(src, evt)
        ujiSemuaCallback(guidata(f)));

uicontrol(f, 'Style', 'pushbutton', 'String', 'Reset', ...
    'Position', [700 480 120 40], 'Callback', @(src, evt) resetCallback(guidata(f)));

uicontrol(f, 'Style', 'pushbutton', 'String', 'Crossval Akurasi', ...
    'Position', [850 480 150 40], 'Callback', @(src, evt) crossvalCallback());

end

function trainingCallback()

    folderSegar = dir('data_training/segarr/*.jpg');
    folderBusuk = dir('data_training/busukk/*.jpg');

    fitur = [];
    label = {};

    for i = 1:length(folderSegar)

        img = imread(fullfile(folderSegar(i).folder, folderSegar(i).name));

```

```

    fitur(end+1,:) = ekstrakFitur(img);

    label{end+1} = 'Segar';

end

for i = 1:length(folderBusuk)

    img = imread(fullfile(folderBusuk(i).folder, folderBusuk(i).name));

    fitur(end+1,:) = ekstrakFitur(img);

    label{end+1} = 'Busuk';

end

modelStruct.model = fitcnb(fitur, label);

modelStruct.numFitur = size(fitur, 2);

save model_nb.mat modelStruct;

msgbox('Model berhasil dilatih dan disimpan!', 'Training');

% Simpan data ekstraksi fitur training ke Excel

header = {'No','Nama

    Citra','Label','Rmean','Gmean','Bmean','Entropy','Contrast','Correlation','Ener

    gy','Homogeneity','Smoothness','Skewness','Intensitas'};

dataTraining = cell(length(label), length(header));

for i = 1:length(label)

    dataTraining{i,1} = i;

```

```

    if i <= length(folderSegar)

        dataTraining{i,2} = folderSegar(i).name;

    else

        dataTraining{i,2} = folderBusuk(i - length(folderSegar)).name;

    end

    dataTraining{i,3} = label{i};

    dataTraining(i,4:end) = num2cell(fitur(i,:));

end

try

    xlswrite('data_training_ekstraksi_fitur.xlsx', [header; dataTraining]);

catch

    warning('Gagal menyimpan file training. Tutup Excel jika sedang terbuka.');
```

end

end

```

function browseCallback(handles)

    [file, path] = uigetfile({'*.jpg;*.png'}, 'Pilih Gambar');

    if isequal(file, 0), return; end

    img = imread(fullfile(path, file));

    axes(handles.axes1); imshow(img); title('Citra Asli');

    gray = rgb2gray(img);
```

```

axes(handles.axes2); imshow(gray); title('Citra Grayscale');

fitur = ekstrakFitur(img);

model_nb = loadOrTrainModel(size(fitur, 2));

hasil = predict(model_nb, fitur);

msgbox(['Hasil Klasifikasi: ' hasil{1}], 'Hasil');

end

function ujiSemuaCallback(handles)

    folderSegar = dir('data_uji/segar/*.jpg');

    folderBusuk = dir('data_uji/busuk/*.jpg');

    namaTest = {};

    labelTest = {};

    fiturTest = [];

    fiturLengkap = [];

    for i = 1:length(folderSegar)

        img = imread(fullfile(folderSegar(i).folder, folderSegar(i).name));

        fitur = ekstrakFitur(img);

        namaTest{end+1} = folderSegar(i).name;

        labelTest{end+1} = 'Segar';

        fiturTest(end+1,:) = fitur;

```



```

        fiturLengkap(end+1,:) = fitur;
    end

    for i = 1:length(folderBusuk)

        img = imread(fullfile(folderBusuk(i).folder, folderBusuk(i).name));

        fitur = ekstrakFitur(img);

        namaTest{end+1} = folderBusuk(i).name;

        labelTest{end+1} = 'Busuk';

        fiturTest(end+1,:) = fitur;

        fiturLengkap(end+1,:) = fitur;
    end

    model_nb = loadOrTrainModel(size(fiturTest,2));

    prediksi = predict(model_nb, fiturTest);

    hasilTabel = cell(length(prediksi), 16);

    benarCount = 0;

    for i = 1:length(prediksi)

        benar = strcmp(prediksi{i}, labelTest{i});

        if benar, benarCount = benarCount + 1; end

        hasilTabel{i,1} = i;
    end

```

```

    hasilTabel{i,2} = namaTest{i};

    hasilTabel{i,3} = labelTest{i};

    hasilTabel{i,4} = prediksi{i};

    hasilTabel{i,5} = ternary(benar, 'Benar', 'Salah');

    hasilTabel(i,6:16) = num2cell(fiturLengkap(i,:));

end

set(handles.uitableHasil, 'Data', hasilTabel);

akurasi = (benarCount / length(prediksi)) * 100;

msgbox(['Akurasi: ' num2str(akurasi, '%.2f') '%'], 'Akurasi');

try

    header = {'No','Nama','Target','Hasil','Keterangan', ...

'Rmean','Gmean','Bmean','Entropy','Contrast','Correlation','Energy','Homogen
eity','Smoothness','Skewness','Intensitas'};

    xlsData = [header; hasilTabel];

    xlswrite('hasil_klasifikasi_apel.xlsx', xlsData);

catch

    warning('Tidak dapat menulis file hasil_klasifikasi_apel.xlsx. Tutup jika
sedang dibuka.');
```

end

```

% Simpan data testing

headerTest = {'No','Nama

    Citra','Label','Rmean','Gmean','Bmean','Entropy','Contrast','Correlation','Ener

    gy','Homogeneity','Smoothness','Skewness','Intensitas'};

dataTesting = cell(length(labelTest), length(headerTest));

for i = 1:length(labelTest)

    dataTesting{i,1} = i;

    dataTesting{i,2} = namaTest{i};

    dataTesting{i,3} = labelTest{i};

    dataTesting(i,4:end) = num2cell(fiturLengkap(i,:));

end

try

    xlswrite('data_testing_ekstraksi_fitur.xlsx', [headerTest; dataTesting]);

catch

    warning('Gagal menyimpan file testing. Tutup Excel jika sedang terbuka.');
```

end

end

```

function crossvalCallback()

    folderSegar = dir('data_training/segarr/*.jpg');

    folderBusuk = dir('data_training/busukk/*.jpg');
```

```

fitur = [];

label = {};

for i = 1:length(folderSegar)

    img = imread(fullfile(folderSegar(i).folder, folderSegar(i).name));

    fitur(end+1,:) = ekstrakFitur(img);

    label{end+1} = 'Segar';

end

for i = 1:length(folderBusuk)

    img = imread(fullfile(folderBusuk(i).folder, folderBusuk(i).name));

    fitur(end+1,:) = ekstrakFitur(img);

    label{end+1} = 'Busuk';

end

label = label(:);

cv = cvpartition(label, 'KFold', 5);

akurasiCV = zeros(cv.NumTestSets,1);

for i = 1:cv.NumTestSets

    trainIdx = training(cv, i);

    testIdx = test(cv, i);

```

```

    model = fitcnb(fitur(trainIdx,:), label(trainIdx));

    pred = predict(model, fitur(testIdx,:));

    labelTest = label(testIdx);

    akurasiCV(i) = sum(strcmp(pred(:), labelTest(:))) / length(pred);

end

msgbox(['Akurasi Rata-rata Crossval 5-Fold: ' num2str(mean(akurasiCV)*100,
    '%.2f) %'], 'Crossval');

end

function resetCallback(handles)

    set(handles.uitableHasil, 'Data', {});

    cla(handles.axes1); title(handles.axes1, 'Citra Asli');

    cla(handles.axes2); title(handles.axes2, 'Citra Grayscale');

    if exist('model_nb.mat', 'file'), delete('model_nb.mat'); end

    if exist('hasil_klasifikasi_apel.xlsx', 'file')

        try, delete('hasil_klasifikasi_apel.xlsx'); catch, end

    end

    if exist('data_training_ekstraksi_fitur.xlsx', 'file')

        try, delete('data_training_ekstraksi_fitur.xlsx'); catch, end

```

```

end

if exist('data_testing_ekstraksi_fitur.xlsx', 'file')

    try, delete('data_testing_ekstraksi_fitur.xlsx'); catch, end

end

msgbox('GUI dan file berhasil di-reset.', 'Reset');

end

function fitur = ekstrakFitur(img)

    img = imresize(img, [100 100]);

    if size(img, 3) == 3

        R = double(img(:,:,1));

        G = double(img(:,:,2));

        B = double(img(:,:,3));

        Rmean = mean(R(:));

        Gmean = mean(G(:));

        Bmean = mean(B(:));

        gray = rgb2gray(img);

    else

        Rmean = 0; Gmean = 0; Bmean = 0;

        gray = img;

    end
end

```

```

gray = double(gray);

entropyVal = entropy(uint8(gray));

contrast = std2(gray);

smoothness = 1 - (1 / (1 + contrast^2));

skew = skewness(gray(:));

intensitas = mean(gray(:));


glcm = graycomatrix(uint8(gray), 'Offset', [0 1]);

stats = graycoprops(glcm);


fitur = [Rmean, Gmean, Bmean, entropyVal, ...
         stats.Contrast, stats.Correlation, stats.Energy, stats.Homogeneity, ...
         smoothness, skew, intensitas];

end


function model_nb = loadOrTrainModel(jumlahFitur)

if exist('model_nb.mat', 'file')

    load model_nb.mat modelStruct;

    if ~isfield(modelStruct, 'numFitur') || modelStruct.numFitur ~= jumlahFitur

        error('Jumlah fitur tidak cocok. Silakan ulangi proses training.', 'Model
Error');

        error('Jumlah fitur tidak cocok.');
```

```
    model_nb = modelStruct.model;

else

    errordlg('Model belum dilatih. Silakan klik "Training Model" dulu.', 'Model
    Error');

    error('Model belum ditemukan.');
```



```
end

end
```



```
function out = ternary(cond, a, b)

    if cond, out = a; else, out = b; end

end
```