%% Real Time Face Detection using MATLAB

% Author: Abubakkar

% Course: DSP Lab

% Date: 28-1-2024

%% Section 1: Basic Image Reading and Display

% Read and display a static image

im = imread('1.jpg');

imshow(im);

title('Original Image');

%% Section 2: Interactive Image Selection

% Use uigetfile to select an image interactively

[file, path] = uigetfile('\*.\*', 'Select image');

if file ~= 0

loc = strcat(path, file);

pic = imread(loc);

figure;

imshow(pic);

title('Selected Image');

end

%% Section 3: Image Processing Operations

% Convert to grayscale and apply thresholding

warning('off');

imgray = rgb2gray(pic);

figure;

imshow(imgray);

title('Grayscale Image');

% Apply automatic thresholding

le = graythresh(pic);

imbw = im2bw(pic, le);

figure;

imshow(imbw);

title('Binary Image (im2bw)');

% Alternative binarization method

imbw2 = imbinarize(imgray, le);

figure;

imshow(imbw2);

title('Binary Image (imbinarize)');

%% Section 4: Save Processed Image

% Save the binary image

imwrite(imbw2, 'bw.jpg', 'jpg');

disp('Binary image saved as bw.jpg');

%% Section 5: Image Manipulation Operations

% Crop image

cpic = imcrop(pic, [100 100 400 400]);

figure;

imshow(cpic);

title('Cropped Image');

% Resize image (scale factor)

rpic = imresize(pic, 0.1);

figure;

imshow(rpic);

title('Resized Image (Scale Factor)');

% Resize image (specific dimensions)

rpic2 = imresize(pic, [500 500]);

figure;

imshow(rpic2);

title('Resized Image (500x500)');

% Flip image horizontally

fpic = flip(pic, 1);

figure;

subplot(1,2,1);

imshow(pic);

title('Original');

subplot(1,2,2);

imshow(fpic);

title('Flipped');

% Rotate image

ropic = imrotate(pic, 30, 'crop');

figure;

imshow(ropic);

title('Rotated Image (30 degrees)');

%% Section 6: Face Detection

% Select image for face detection

[file, path] = uigetfile('\*.\*', 'Select image for face detection');

if file ~= 0

loc = strcat(path, file);

pic = imread(loc);

pic2 = rgb2gray(pic);

% Create face detector

ff = vision.CascadeObjectDetector();

bbox = step(ff, pic2);

% Annotate detected faces

dd = insertObjectAnnotation(pic, 'Rectangle', bbox, 'Face');

% Detect corner features in face region

if ~isempty(bbox)

pts = detectMinEigenFeatures(pic2, 'ROI', bbox);

dd = insertMarker(dd, pts, '+', 'color', 'green');

end

figure;

imshow(dd);

title('Face Detection with Corner Points');

if ~isempty(bbox)

hold on;

plot(pts.Location(:,1), pts.Location(:,2), 'g.');

hold off;

end

end

%% Section 7: Mouth Detection

% Detect mouth

if exist('pic2', 'var')

fm = vision.CascadeObjectDetector('Mouth');

fm.MergeThreshold = 110;

bbox\_mouth = step(fm, pic2);

dd\_mouth = insertObjectAnnotation(pic, 'Rectangle', bbox\_mouth, 'Mouth');

figure;

imshow(dd\_mouth);

title('Mouth Detection');

end

%% Section 8: Nose Detection

% Detect nose

if exist('pic2', 'var')

fn = vision.CascadeObjectDetector('Nose');

bbox\_nose = step(fn, pic2);

dd\_nose = insertObjectAnnotation(pic, 'Rectangle', bbox\_nose, 'Nose');

figure;

imshow(dd\_nose);

title('Nose Detection');

end

%% Section 9: Eye Detection

% Detect eyes

if exist('pic2', 'var')

fe = vision.CascadeObjectDetector('RightEye', 'MergeThreshold', 40);

bbox\_eye = step(fe, pic2);

dd\_eye = insertObjectAnnotation(pic, 'Rectangle', bbox\_eye, 'Eye');

figure;

imshow(dd\_eye);

title('Eye Detection');

end

%% Section 10: Upper Body Detection

% Detect upper body

if exist('pic2', 'var')

fb = vision.CascadeObjectDetector('UpperBody', 'MergeThreshold', 5);

bbox\_body = step(fb, pic2);

dd\_body = insertObjectAnnotation(pic, 'Rectangle', bbox\_body, 'Body');

figure;

imshow(dd\_body);

title('Upper Body Detection');

end

%% Section 11: Basic Webcam Setup

% Initialize webcam (uncomment to use)

% web = webcam('HD WebCam');

% preview(web);

% pause(2);

% pp = snapshot(web);

% imshow(pp);

% pause(2);

% clear('web');

%% Section 12: Simple Webcam Loop

% Basic webcam display loop (uncomment to use)

% web = webcam();

% while true

% pic = snapshot(web);

% imshow(pic);

% pause(0.01);

% % Press Ctrl+C to stop

% end

%% Section 13: Real-time Face Detection with Webcam

% Real-time face detection using webcam

function realTimeFaceDetection()

clc;

close all;

try

% Initialize webcam

li = webcam();

% Take initial snapshot

im = snapshot(li);

% Create face detector (corrected to use 'FrontalFaceCART' instead of 'Mouth')

dete = vision.CascadeObjectDetector('FrontalFaceCART', 'MergeThreshold', 4);

% Create figure for display

figure('Name', 'Real-time Face Detection', 'NumberTitle', 'off');

fprintf('Starting real-time face detection. Press Ctrl+C to stop.\n');

while true

% Capture frame

im = snapshot(li);

im2 = rgb2gray(im);

% Detect faces

bb = step(dete, im2);

% Annotate detected faces

if ~isempty(bb)

im\_annotated = insertObjectAnnotation(im, 'rectangle', bb, 'Face');

else

im\_annotated = im;

end

% Display result

imshow(im\_annotated);

title(sprintf('Real-time Face Detection - %d face(s) detected', size(bb, 1)));

drawnow;

% Small pause to prevent overwhelming the system

pause(0.03);

end

catch ME

if exist('li', 'var')

clear li;

end

fprintf('Error: %s\n', ME.message);

end

end

%% Section 14: Enhanced Real-time Face Detection

% Enhanced version with better error handling

function enhancedRealTimeFaceDetection()

clc;

close all;

try

% Check if webcam is available

camList = webcamlist;

if isempty(camList)

error('No webcam found. Please connect a webcam and try again.');

end

% Initialize webcam

cam = webcam(1);

% Create face detector

faceDetector = vision.CascadeObjectDetector('FrontalFaceCART');

faceDetector.MergeThreshold = 4;

faceDetector.MinSize = [60 60];

% Create figure

hFig = figure('Name', 'Enhanced Real-time Face Detection', ...

'NumberTitle', 'off', ...

'CloseRequestFcn', @(src,evt)cleanupAndClose(src,evt,cam));

fprintf('Enhanced real-time face detection started.\n');

fprintf('Close the figure window to stop detection.\n');

while ishandle(hFig)

% Capture frame

frame = snapshot(cam);

% Convert to grayscale for detection

grayFrame = rgb2gray(frame);

% Detect faces

bboxes = step(faceDetector, grayFrame);

% Annotate detected faces

if ~isempty(bboxes)

annotatedFrame = insertObjectAnnotation(frame, 'rectangle', ...

bboxes, 'Face', 'LineWidth', 2, 'Color', 'red');

else

annotatedFrame = frame;

end

% Display result

imshow(annotatedFrame);

title(sprintf('Enhanced Face Detection - %d face(s) detected', size(bboxes, 1)));

drawnow;

% Control frame rate

pause(0.03);

end

catch ME

if exist('cam', 'var')

clear cam;

end

fprintf('Error: %s\n', ME.message);

end

end

% Cleanup function for enhanced detection

function cleanupAndClose(src, evt, cam)

try

clear cam;

catch

end

delete(src);

end

%% Usage Instructions

% To run real-time face detection, uncomment one of the following lines:

% realTimeFaceDetection();

% enhancedRealTimeFaceDetection();

fprintf('Face detection code loaded successfully!\n');

fprintf('To run real-time detection, call:\n');

fprintf(' realTimeFaceDetection() - Basic version\n');

fprintf(' enhancedRealTimeFaceDetection() - Enhanced version\n');