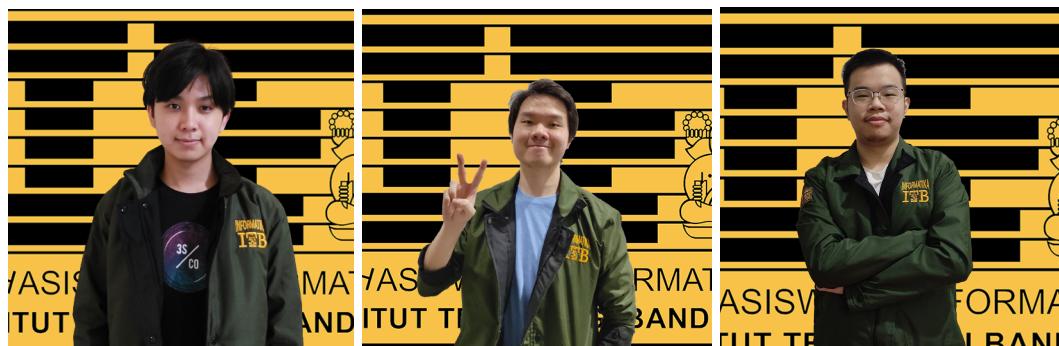


LAPORAN TUGAS

Automatic Plate Number Recognition

Mata Kuliah IF4071 Interpretasi dan Pengolahan Citra

Dosen Pengampu: Rinaldi Munir



Disusun Oleh:

Christopher J.W. 13519006

Billy Julius 13519094

Girvin Junod 13519096

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

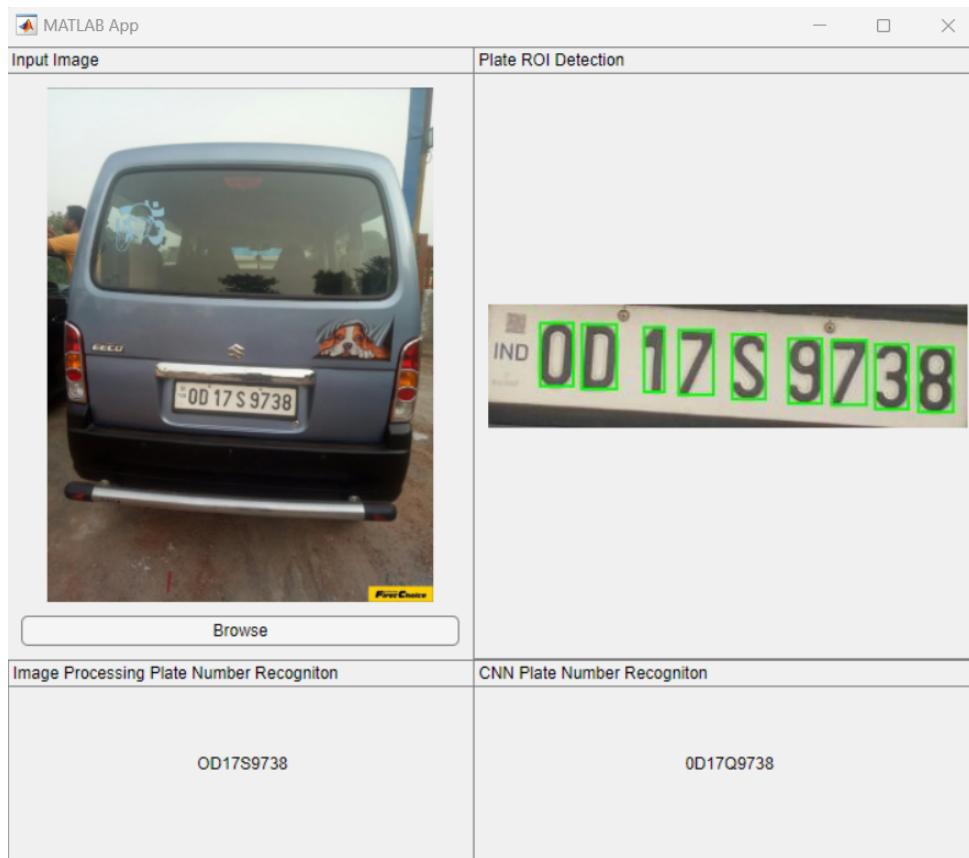
2022

Daftar Isi

1. Screenshot GUI	2
2. Kode Program	3
2.1. Plate ROI Detection	3
2.2. Pengenalan Karakter dengan Teknik Pengolahan Citra	4
2.3. Convolutional Neural Network (CNN)	7
2.4. Character Detection with CNN	8
2.5. Main Program	9
3. Contoh Hasil Eksekusi Program	9
4. Analisis Hasil Program	16
5. Kesimpulan	17
6. Komentar dan Refleksi	17
7. Alamat Github Program	18

1. Screenshot GUI

Berikut adalah tampilan GUI dari aplikasi yang dibangun. Pada bagian kiri atas, terdapat tombol untuk memasukkan citra input dan menampilkannya ke layar. Di bagian kanan atas, terdapat gambar hasil deteksi ROI berupa plat kendaraan dan disertai dengan *box-bounding* yang menandakan karakter-karakter yang terdeteksi. Hasil pengenalan plat kendaraan terdapat di bagian bawah dengan bagian kiri adalah hasil pengenalan dengan menggunakan pengolahan citra alami dan di bagian kanan adalah hasil pengenalan dengan menggunakan *convolutional neural network*.



2. Kode Program

2.1. Plate ROI Detection

Proses pendekripsi ROI plat kendaraan dilakukan dengan menggunakan fungsi `detectPlate`. Fungsi tersebut menerima input citra kendaraan dengan plat bermotor lalu mengembalikan ROI dari plat yang dicari untuk digunakan pada tahap selanjutnya yaitu pendekripsi karakter. Langkah pertama untuk melakukan pendekripsi plat adalah dengan menerapkan pemerataan histogram pada citra yang telah dikonversi menjadi citra *grayscale* terlebih dahulu untuk memperbaiki kontras citra. Selanjutnya pendekripsi tepi pada citra akan dilakukan dengan menggunakan operator Sobel. Hasil deteksi tepi tersebut merupakan sebuah citra biner yang terdiri dari sekumpulan garis yang menandakan tepi objek. Dilatasi diterapkan pada citra biner tersebut untuk mempertegas garis-garis tepi, serta diterapkan juga pengisian pada area di dalam tepi menghasilkan *mask* citra biner. *Mask* tersebut digunakan untuk mencari *bounding box* di dalam citra asli. Program kemudian akan membandingkan tiap *bounding box* sesuai dengan ukuran dan rasio panjang dan lebar dari *bounding box* dan yang terpilih akan digunakan untuk memotong citra sehingga hanya menyisakan daerah ROI plat saja.

Deteksi ROI Plat

```
function plate = detectPlate(image)

    % Convert input image to grayscale
    grayImg = im2gray(image);

    % Apply histogram equalization
    grayImg = histeq(grayImg);

    % Apply edge detection using gradient operator
    edgeImg = edge(grayImg, 'sobel');

    % Apply line dilatation and gap filling
    se90 = strel('line', 3, 90);
    se0 = strel('line', 3, 0);

    dilatedImg = imdilate(edgeImg, [se90 se0]);
    filledImg = imfill(dilatedImg, 'holes');

    clearedBorder = imclearborder(filledImg, 4);

    % Erode mask
    se = strel('diamond', 3);
    mask =imerode(clearedBorder, se);
    mask =imerode(mask, se);

    figure, imshow(labeloverlay(image, mask));

    % Get plate region
```

```

regions = regionprops(mask, 'BoundingBox', 'Area', 'Image');
boundingBox = regions.BoundingBox;
max = regions.Area;

for i = 1:numel(regions)
    [y, x] = size(regions(i).Image);
    if regions(i).Area > max && x > (3 * y) && x < (5 * y)
        max = regions(i).Area;
        boundingBox = regions(i).BoundingBox;
    end
end

plate = imcrop(image, boundingBox);

```

2.2. Pengenalan Karakter dengan Teknik Pengolahan Citra

Pengenalan karakter pada plat nomor dilakukan melalui metode *template matching* karakter dengan *template* karakter yang telah dibuat. Langkah pengenalan karakter adalah pertama membuat data *template* karakter untuk kebutuhan *template matching*. Ini dapat dilakukan dengan mengumpulkan sekumpulan gambar karakter berukuran sama dan menyimpannya dalam file .mat. Lalu, dilakukan segmentasi gambar plat nomor untuk mendapatkan gambar dari masing-masing karakter yang ada. Dari setiap gambar karakter ini, akan dicari gambar *template* karakter mana yang paling mirip terhadap gambar karakter tersebut. Setiap *template* karakter ini juga memiliki label untuk menandakan gambar tersebut melambangkan karakter apa sehingga dapat diambil nilai label dari gambar *template* paling mirip. Selain deteksi karakter, dikembalikan juga gambar plat nomor dengan *bounding box* karakter yang terdeteksi.

Berikut adalah potongan kode untuk membuat data *template* karakter, segmentasi karakter plat nomor, mengambil *bounding box* dan gambar karakter, dan pengenalan karakter melalui *template matching*.

Membuat data *template* Karakter

```

% Script file for loading templates

% Directory
templateDir = '../img/charTemplate';

% All the template chars
% Some chars have multiple versions to account for different fonts
A = imread(fullfile(templateDir, 'A.bmp'));
B = imread(fullfile(templateDir, 'B.bmp'));
C = imread(fullfile(templateDir, 'C.bmp'));
D = imread(fullfile(templateDir, 'D.bmp'));
E = imread(fullfile(templateDir, 'E.bmp'));

```

```

F = imread(fullfile(templateDir, 'F.bmp'));
G = imread(fullfile(templateDir, 'G.bmp'));
H = imread(fullfile(templateDir, 'H.bmp'));
I = imread(fullfile(templateDir, 'I.bmp'));
I2 = imread(fullfile(templateDir, 'I_2.bmp'));
J = imread(fullfile(templateDir, 'J.bmp'));
K = imread(fullfile(templateDir, 'K.bmp'));
L = imread(fullfile(templateDir, 'L.bmp'));
M = imread(fullfile(templateDir, 'M.bmp'));
M2 = imread(fullfile(templateDir, 'M_2.bmp'));
N = imread(fullfile(templateDir, 'N.bmp'));
O = imread(fullfile(templateDir, 'O.bmp'));
P = imread(fullfile(templateDir, 'P.bmp'));
Q = imread(fullfile(templateDir, 'Q.bmp'));
R = imread(fullfile(templateDir, 'R.bmp'));
S = imread(fullfile(templateDir, 'S.bmp'));
T = imread(fullfile(templateDir, 'T.bmp'));
U = imread(fullfile(templateDir, 'U.bmp'));
V = imread(fullfile(templateDir, 'V.bmp'));
W = imread(fullfile(templateDir, 'W.bmp'));
X = imread(fullfile(templateDir, 'X.bmp'));
Y = imread(fullfile(templateDir, 'Y.bmp'));
Z = imread(fullfile(templateDir, 'Z.bmp'));

one = imread(fullfile(templateDir, '1.bmp'));
one2 = imread(fullfile(templateDir, '1_2.bmp'));
two = imread(fullfile(templateDir, '2.bmp'));
three = imread(fullfile(templateDir, '3.bmp'));
four = imread(fullfile(templateDir, '4.bmp'));
five = imread(fullfile(templateDir, '5.bmp'));
six = imread(fullfile(templateDir, '6.bmp'));
seven = imread(fullfile(templateDir, '7.bmp'));
eight = imread(fullfile(templateDir, '8.bmp'));
nine = imread(fullfile(templateDir, '9.bmp'));
zero = imread(fullfile(templateDir, '0.bmp'));
zero2 = imread(fullfile(templateDir, '0_2.bmp'));

letters = [A B C D E F G H I I2 J K L M M2 N O P Q R S T U V W X Y Z];
numbers = [one one2 two three four five six seven eight nine zero zero2];
template = [letters numbers];

% Save template to a .mat file
save ('TemplateFile', 'template')

clear all

```

Segmentasi karakter pada gambar plat nomor

```

function ret = segmentPlate(plateImg)
    %SEGMENTPlate function to segment the plate image into letters/numbers
    % plateImg: the image of the plate
    % ret: the segmented plate letter/number images

    % apply thresholding using imbinarize
    binImg = imbinarize(plateImg);

    % Remove small objects

```

```

cleanImg = bwareaopen(binImg, round(0.001 * numel(binImg)));

% if the white part of the image is bigger than the black part, invert
the image
% Segment image and get bounding box
if bwarea(cleanImg) < bwarea(~cleanImg)
    segments = regionprops(cleanImg, 'Image', 'BoundingBox');
else
    segments = regionprops(~cleanImg, 'Image', 'BoundingBox');
end

ret = segments
end

```

Mengambil *bounding box* dan gambar karakter tersegmentasi

```

function [listImg, listBB] = getSegmentedChar(segment, imgSize)
    %getSegmentedChar - Get segmented character from segmented image

    listImg = {};
    listBB = {};

    % Get Image height
    h = imgSize(1);

    % for each segment, get the bounding box and the char image
    for i = 1:length(segment)
        seg = segment(i);

        [y, x] = size(seg.Image);

        % Filter out segment that doesn't fit the criteria
        % Segment should be less than 80% of original image height, more than
        20% of original image height
        % and the length should be less than the height
        if y < (0.8 * h) && y > (0.2 * h) && x <= y

            img = seg.Image;
            bb = seg.BoundingBox;
            listImg{end + 1} = img;
            listBB{end + 1} = bb;
        end
    end
end

```

Pengenalan karakter melalui *template matching*

```

function ret = detectChar(letterImg)
    % Load template images
    load templateFile;

    % Labels for template images

```

```

label = [
    'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'I', 'J', 'K',
    'L', 'M', 'M', ...
    'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X',
    'Y', 'Z', ...
    '1', '1', '2', '3', '4', '5', '6', '7', '8', '9', '0', '0'
];
labelCount = size(label, 2);

% Image height
y = 24;

% Resize img to fit template
letterImg = imresize(letterImg, [42 y]);

listcorr = [];

for i = 1:labelCount
    % Get template and binarize
    temp = imbinarize(template(:, ((i - 1) * y) + 1:i * y));

    % Get correlation
    corr = corr2(temp, letterImg);

    % Add correlation to list
    listcorr = [listcorr corr];
end

% Get max correlation
idx = find(listcorr == max(listcorr));

% Get label for letter
ret = label(idx);
end

```

2.3. Convolutional Neural Network (CNN)

Pengenalan plat kendaraan dengan *Convolutional Neural Network* (CNN) menggunakan model *pre-trained* dari **VGG-19** dan dilatih kembali dengan menggunakan data [AI Indian License Plate Recognition Data](#) yang berisikan gambar digit-digit dari plat nomor dengan kelasnya masing-masing, yaitu alphanumerik dari 0 sampai 9 dan A sampai Z.

Berikut adalah potongan kode program yang mendefinisikan penyusunan model berdasarkan model dasar VGG-19 dan menambahkan sebuah *fully connected layer* dengan 128 unit dan disertai dengan *fully connected layer* terakhir sebagai *output layer* dengan fungsi aktivasi *softmax* dan 36 unit sesuai dengan jumlah alphanumerik.

```
def Model(learning_rate):
```

```

base_model = VGG19(include_top=False, input_tensor=Input(shape =
(32,32,3)))

model = Sequential()
for layer in base_model.layers:
    model.add(layer)

model.add(Flatten())
model.add(Dense(128, activation="relu"))
model.add(Dense(36, activation="softmax"))

loss = "categorical_crossentropy"
optimizer = Adam(learning_rate=learning_rate)
metrics = ["accuracy"]
model.compile(
    loss = loss,
    optimizer = optimizer,
    metrics = metrics
)

return model

```

Model kemudian dilatih dengan *learning rate* sebesar 2e-5 dan 30 epochs. Hasilnya didapatkan bahwa model memiliki akurasi validasi sebesar 97.69%. Namun, hasil ini tampak bersifat *overfit*. Hal ini dikarenakan model memiliki akurasi yang cukup tinggi saat pelatihan namun saat dicobakan dengan gambar plat kendaraan asli, model tidak menampilkan hasil yang cukup baik. Contoh hasil prediksi dari model dapat dilihat pada bagian selanjutnya dibawah.

```

Epoch 30/30
864/864 [=====] - 20s 23ms/step - loss: 0.1485 - accuracy: 0.9664 - val_loss: 0.0388 - val_accuracy: 0.9769

```

2.4. Character Detection with CNN

Dengan menggunakan model yang telah dilatih di tahap sebelumnya, pendekripsi karakter kemudian dilakukan dengan melakukan prediksi dengan model yang dimuat sebelumnya dan mengambil output dari prediksi tersebut yang berupa karakter alphanumerik 0 sampai 9 dan A sampai Z.

```

function ret = detectCharCNN(net, letterImg)
% Labels
label = [
    '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', ...
    'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M',
    'M', ...
    'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z', ...
];
% Reformat the image as 3 channels
rgbImage = cat(3, letterImg, letterImg, letterImg);

```

```

% Resize img
letterImg = imresize(rgbImage, [32 32]);
% Predict the letter
prediction = predict(net, letterImg);
[~, argmax] = max(prediction);

ret = label(argmax);
end

```

2.5. Main Program

Berikut adalah cuplikan kode untuk penggunaan fungsi-fungsi di atas dalam pengenalan karakter dari citra kendaraan dengan plat nomor. Pada fungsi ini dikembalikan karakter-karakter yang terdeteksi dalam citra dan gambar plat nomor beserta *bounding box* untuk menandai karakter.

```

function [letters, bbImg] = detectCharFromPlate(img, method, net)
    % Turn image into grayscale
    grayimg = im2gray(img);

    % Segment plate into characters and bounding boxes
    ret = segmentPlate(grayimg);
    [chars, bb] = getSegmentedChar(ret, size(grayimg));

    letters = [];
    bbImg = img;

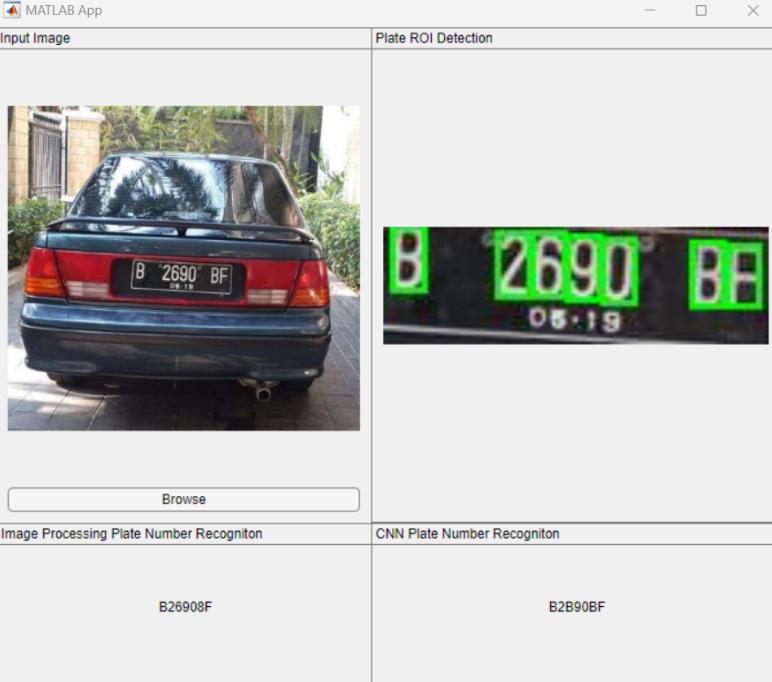
    % For each character, do a template matching for char recognition
    for i = 1:length(chars)
        if (method == 'CNN')
            letters = [letters, detectCharCNN(net, chars{i})];
        else
            letters = [letters, detectChar(chars{i})];
        end

        % Draw bounding box on original image
        BB = bb{i};
        bbImg = insertShape(bbImg, 'Rectangle', [BB(1), BB(2), BB(3), BB(4)], 
        LineWidth = 2, Color = 'green');
    end

```

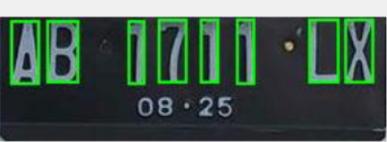
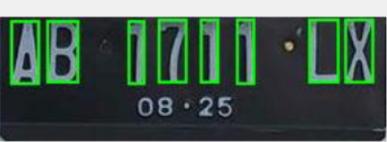
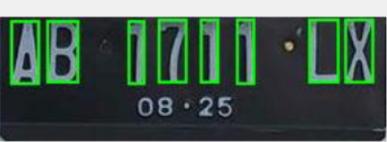
3. Contoh Hasil Eksekusi Program

Berikut adalah contoh hasil eksekusi program dengan menggunakan beberapa gambar mobil yang diambil dari internet.

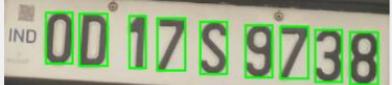
Input Gambar	Hasil Eksekusi	Hasil <i>Image Processing</i>	Hasil CNN
 B2690BF	 <p>MATLAB App</p> <p>Input Image</p> <p>Plate ROI Detection</p> <p>Browse</p> <p>Image Processing Plate Number Recogniton</p> <p>CNN Plate Number Recogniton</p> <p>B26908F</p> <p>B2B90BF</p>	B26908F (86%)	B2690 BF (100%)
 BG9866NS	 <p>MATLAB App</p> <p>Input Image</p> <p>Plate ROI Detection</p> <p>Browse</p> <p>Image Processing Plate Number Recogniton</p> <p>CNN Plate Number Recogniton</p> <p>BG9866NS</p> <p>BG9866MQ</p>	8G9866NS (89%)	BG986 6MQ (78%)

 <p>B360LU</p>	<p>MATLAB App</p> <table border="1"> <thead> <tr> <th>Input Image</th><th>Plate ROI Detection</th></tr> </thead> <tbody> <tr> <td></td><td></td></tr> <tr> <td><input type="button" value="Browse"/></td><td></td></tr> <tr> <td>Image Processing Plate Number Recogniton</td><td>CNN Plate Number Recogniton</td></tr> <tr> <td>LB360LU</td><td>KB360KS</td></tr> </tbody> </table>	Input Image	Plate ROI Detection			<input type="button" value="Browse"/>		Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	LB360LU	KB360KS	<p>LB360LU (86%)</p>	<p>KB360 KS (57%)</p>
Input Image	Plate ROI Detection												
<input type="button" value="Browse"/>													
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton												
LB360LU	KB360KS												
 <p>B234KIL</p>	<p>MATLAB App</p> <table border="1"> <thead> <tr> <th>Input Image</th> <th>Plate ROI Detection</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> <tr> <td><input type="button" value="Browse"/></td> <td></td> </tr> <tr> <td>Image Processing Plate Number Recogniton</td> <td>CNN Plate Number Recogniton</td> </tr> <tr> <td>B234K1L</td> <td>B23XJIK</td> </tr> </tbody> </table>	Input Image	Plate ROI Detection			<input type="button" value="Browse"/>		Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	B234K1L	B23XJIK	<p>B234K1L (86%)</p>	<p>B23KJI K (57%)</p>
Input Image	Plate ROI Detection												
<input type="button" value="Browse"/>													
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton												
B234K1L	B23XJIK												

 <p>B1234WLG</p>	<p>Matlab App</p> <table border="1"> <thead> <tr> <th>Input Image</th><th>Plate ROI Detection</th></tr> </thead> <tbody> <tr> <td></td><td></td></tr> <tr> <td><input type="button" value="Browse"/></td><td></td></tr> <tr> <td>Image Processing Plate Number Recognition</td><td>CNN Plate Number Recognition</td></tr> <tr> <td>B1234WLG</td><td>B1234UKG</td></tr> </tbody> </table>	Input Image	Plate ROI Detection			<input type="button" value="Browse"/>		Image Processing Plate Number Recognition	CNN Plate Number Recognition	B1234WLG	B1234UKG	<p>B1234WL G (100%)</p>	<p>B1234 UKG (75%)</p>
Input Image	Plate ROI Detection												
													
<input type="button" value="Browse"/>													
Image Processing Plate Number Recognition	CNN Plate Number Recognition												
B1234WLG	B1234UKG												
 <p>H9091TE</p>	<p>Matlab App</p> <table border="1"> <thead> <tr> <th>Input Image</th><th>Plate ROI Detection</th></tr> </thead> <tbody> <tr> <td></td><td></td></tr> <tr> <td><input type="button" value="Browse"/></td><td></td></tr> <tr> <td>Image Processing Plate Number Recognition</td><td>CNN Plate Number Recognition</td></tr> <tr> <td>H9091TE</td><td>H9091RE</td></tr> </tbody> </table>	Input Image	Plate ROI Detection			<input type="button" value="Browse"/>		Image Processing Plate Number Recognition	CNN Plate Number Recognition	H9091TE	H9091RE	<p>H9091TE (100%)</p>	<p>H9091 RE (86%)</p>
Input Image	Plate ROI Detection												
													
<input type="button" value="Browse"/>													
Image Processing Plate Number Recognition	CNN Plate Number Recognition												
H9091TE	H9091RE												

 <p>B1725SAH</p>	<p>MATLAB App</p> <table border="1"> <thead> <tr> <th>Input Image</th><th>Plate ROI Detection</th></tr> </thead> <tbody> <tr> <td></td><td></td></tr> <tr> <td><input type="button" value="Browse"/></td><td></td></tr> <tr> <td>Image Processing Plate Number Recogniton</td><td>CNN Plate Number Recogniton</td></tr> <tr> <td>B1725SA8</td><td>B1725QAH</td></tr> </tbody> </table>	Input Image	Plate ROI Detection			<input type="button" value="Browse"/>		Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	B1725SA8	B1725QAH	<p>B1725SA8 (88%)</p>	<p>B1725 QAH (88%)</p>
Input Image	Plate ROI Detection												
													
<input type="button" value="Browse"/>													
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton												
B1725SA8	B1725QAH												
 <p>AB1711LX</p>	<p>MATLAB App</p> <table border="1"> <thead> <tr> <th>Input Image</th><th>Plate ROI Detection</th></tr> </thead> <tbody> <tr> <td></td><td></td></tr> <tr> <td><input type="button" value="Browse"/></td><td></td></tr> <tr> <td>Image Processing Plate Number Recogniton</td><td>CNN Plate Number Recogniton</td></tr> <tr> <td>AB1Y11LX</td><td>AB1711KV</td></tr> </tbody> </table>	Input Image	Plate ROI Detection			<input type="button" value="Browse"/>		Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	AB1Y11LX	AB1711KV	<p>AB1Y11LX (88%)</p>	<p>AB171 1KV (75%)</p>
Input Image	Plate ROI Detection												
													
<input type="button" value="Browse"/>													
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton												
AB1Y11LX	AB1711KV												

 <p>D1318ADH</p>	<p>Input Image</p>  <p>Browse</p> <table border="1"> <tr> <td>Image Processing Plate Number Recogniton</td> <td>CNN Plate Number Recogniton</td> </tr> </table> <p>D1318ADH</p>	Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	<p>Plate ROI Detection</p> 	<p>D1318ADH (100%)</p>	<p>D131B MDH (88%)</p>
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton					
 <p>B1341NRQ</p>	<p>Input Image</p>  <p>Browse</p> <table border="1"> <tr> <td>Image Processing Plate Number Recogniton</td> <td>CNN Plate Number Recogniton</td> </tr> </table> <p>B1341NRQ</p>	Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	<p>Plate ROI Detection</p> 	<p>B1341NRU (88%)</p>	<p>B13X1 MPO (50%)</p>
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton					

 GA13BYZ	<p>Input Image</p>  <p>Plate ROI Detection</p>  <p>Browse</p> <table border="1"> <tr> <td>Image Processing Plate Number Recogniton</td><td>CNN Plate Number Recogniton</td></tr> <tr> <td>CA538YZ</td><td>GAH3BWX</td></tr> </table>	Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	CA538YZ	GAH3BWX	CA538YZ (57%)	GAH3BWX (57%)
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton						
CA538YZ	GAH3BWX						
 OD17S9738	<p>Input Image</p>  <p>Plate ROI Detection</p>  <p>Browse</p> <table border="1"> <tr> <td>Image Processing Plate Number Recogniton</td> <td>CNN Plate Number Recogniton</td> </tr> <tr> <td>OD17S9738</td> <td>OD17Q9738</td> </tr> </table>	Image Processing Plate Number Recogniton	CNN Plate Number Recogniton	OD17S9738	OD17Q9738	OD17S973 8 (100%)	OD17 Q9738 (89%)
Image Processing Plate Number Recogniton	CNN Plate Number Recogniton						
OD17S9738	OD17Q9738						

 HP12F0957	MATLAB App Input Image Plate ROI Detection  <input type="button" value="Browse"/> Image Processing Plate Number Recogniton CNN Plate Number Recogniton HP12FO952 HN12F0957	HP12FO95 2 (78%)	HN12F 0957 (89%)
------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------	------------------------

4. Analisis Hasil Program

Dari contoh hasil eksekusi program di atas, dapat dilihat bahwa dengan teknik pengolahan citra, didapatkan hasil yang lebih konsisten dan akurat. Adapun perbedaan pengenalan karakter disebabkan oleh perbedaan cara dalam mengenali karakter dalam plat kendaraan antara pendekatan pengolahan citra alami dan *convolutional neural network* (CNN). Adapun hasil rata-rata akurasi kedua metode dari percobaan yang dilakukan adalah 88,2% untuk pengenalan karakter plat kendaraan dengan teknik pengolahan citra dan 76,1%.

Untuk pendekatan dengan pengolahan citra, beberapa karakter salah dikenali akibat kemiripan karakter tersebut dengan karakter lain seperti untuk karakter 0 dan O pada plat kendaraan. Masalah juga sering muncul dari perbedaan *font* pada plat kendaraan dengan *font* yang digunakan pada gambar *template* karakter. Ini memunculkan beberapa masalah karena untuk beberapa huruf perbedaannya bisa jadi cukup signifikan. Kami telah mencoba untuk mengatasi masalah ini dengan menambah gambar *template* karakter namun tetap ada beberapa kasus yang tetap terjadi kesalahan.

Kemudian untuk CNN, terdapat beberapa karakter yang sering salah dikenali, seperti 4 yang sering dikenali sebagai X dan R yang sering dikenali sebagai P.

Selain itu, tampak bahwa model CNN yang digunakan lebih mampu dalam mengenali digit angka dibandingkan dengan huruf. Hal ini mungkin disebabkan oleh data digit angka yang digunakan untuk pelatihan cukup bervariasi dan berbeda antar satu sama lain relatif dibandingkan dengan huruf.

Untuk pendekripsi ROI plat terdapat beberapa batasan yang menyebabkan kesalahan pada deteksi yaitu warna kendaraan dan plat yang sama tanpa pembatas area yang jelas menyebabkan pendekripsi tepi untuk daerah plat gagal untuk dideteksi. Kemudian terdapat juga kesalahan pada pemilihan bounding box dari beberapa kandidat karena kriteria pemilihan bounding box hanya didasari oleh ukuran bounding box terbesar yang sesuai dengan rasio panjang dan lebar tertentu. Lalu pendekripsi ROI plat hanya efektif terbatas pada plat berwarna hitam atau putih.

Program kami berhasil mengenali karakter dari citra kendaraan yang tidak ideal seperti citra yang miring atau kabur. Performa program kami untuk citra-citra yang tidak ideal ini cukup baik dan dapat dilihat pada beberapa *screenshot* hasil percobaan di atas.

5. Kesimpulan

Melalui tugas ini, telah berhasil dikembangkan sebuah program MATLAB yang mampu mengenali nomor plat kendaraan dengan dua cara, yaitu dengan pengolahan citra dan menggunakan *convolutional neural network* (CNN). Berdasarkan analisis hasil program, disimpulkan bahwa pendekatan dengan teknik pengolahan citra memiliki konsistensi dan akurasi yang lebih baik dibandingkan dengan pendekatan CNN.

6. Komentar dan Refleksi

Tugas IF4073 ini lebih menarik dan menantang dari tugas-tugas sebelumnya. Dari tugas ini juga jadi lebih terlihat aplikasi hal yang telah dipelajari dalam mata kuliah ini di dunia nyata.

Masih terdapat banyak hal yang dapat dikembangkan dan diperbaiki dari program yang kami buat. Untuk pengenalan karakter dengan pengolahan citra, dapat dikembangkan lagi pada sisi pengenalan karakternya, terutama untuk karakter-karakter yang mirip. Untuk pengenalan digit berbasis *convolutional neural network*, pengembangan dapat dilakukan dengan memperkaya variasi data yang digunakan untuk pelatihan. Hal ini didasarkan pada jenis tulisan yang digunakan dalam plat kendaraan sangatlah bervariasi serta variasi dari data

yang digunakan untuk pelatihan masih relatif rendah, maka model dapat dikembangkan lebih jauh lagi dengan memperkaya variasi data yang digunakan untuk pelatihan.

7. Alamat Github Program

Github: <https://github.com/billyjuliux/LicensePlateRecognition>

Notebook for CNN Model Experiment:

[https://colab.research.google.com/drive/1Gw5lOG6gBHRVVWjIXU0kxsUr7J2tGY
SY?usp=sharing](https://colab.research.google.com/drive/1Gw5lOG6gBHRVVWjIXU0kxsUr7J2tGYSY?usp=sharing)