Kelompok 5

DETEKSI DAN PENGENALAN KTP

PENGOLAHAN CITRA DAN VISI KOMPUTER

KUIS 2

Anggota Kelompok

Abdul Basith Bahi 01

M. Dzaka Murran Rusid 21

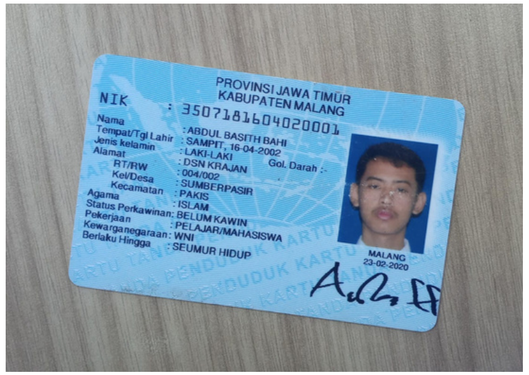
Bagus Rezky Adhyaksa 07

Sely Ruli Amanda 23

Maulana Arif Wijaya 16

KELOMPOK 5

CITRA KTP



KELOMPOK 5

Preprocessing

Image

```
# Iterasi pada gambar-gambar KTP
for i, ktp_file in enumerate(ktp_files):
    ktp_path = os.path.join(folder_path, ktp_file)
    ktp_image = cv2.imread(ktp_path)

    blur = cv2.blur(ktp_image, (5,5))
    _, mask = cv2.threshold(blur, 180, 255, cv2.THRESH_BINARY)
    mask[:,:,1] = 0
    mask[:,:,2] = 0

    lt = 50
    edges = cv2.Canny(mask, lt, lt * 3)

    kernel = np.ones((5, 5), np.uint8)

    dilatation_dst = cv2.dilate(edges, kernel , iterations=4)
    erosion = cv2.erode(dilatation_dst,kernel,iterations = 3)

    # Deteksi tepi menggunakan Canny
    edges = cv2.Canny(erosion, 50, 150)

    plt.figure(figsize=(15, 12))

    plt.subplot(3, 2, 1)
    plt.imshow(cv2.cvtColor(ktp_image, cv2.COLOR_BGR2RGB))
    plt.title('Original Image')
    plt.axis('off')

    plt.subplot(3, 2, 2)
    plt.imshow(cv2.cvtColor(blur, cv2.COLOR_BGR2RGB))
    plt.title('Blurred Image')
    plt.axis('off')

    plt.subplot(3, 2, 3)
    plt.imshow(cv2.cvtColor(mask, cv2.COLOR_BGR2RGB))
    plt.title('Thresholded Image')
    plt.axis('off')

    plt.subplot(3, 2, 4)
    plt.imshow(cv2.cvtColor(erosion, cv2.COLOR_BGR2RGB))
    plt.title('Dilated and Eroded Image')
    plt.axis('off')

    plt.subplot(3, 2, 5)
    plt.imshow(cv2.cvtColor(edges, cv2.COLOR_BGR2RGB))
    plt.title('Canny')
    plt.axis('off')

    plt.show()
```

KELOMPOK 5

Hasil preprocessing

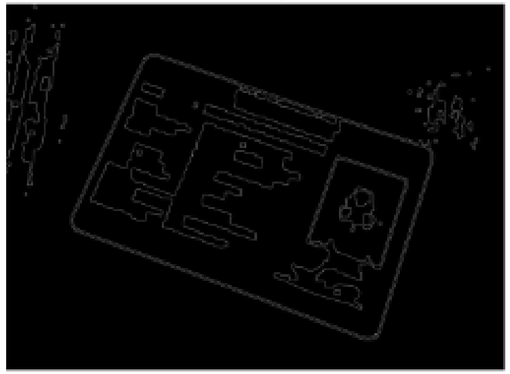
Image

Original Image

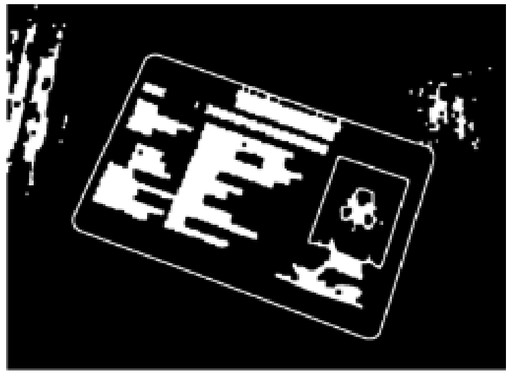
Blurred Image

Thresholded Image

Canny



Dilated and Eroded Image



KELOMPOK 5

Lokalisasi Image

```
contours, img = cv2.findContours(edges, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_NONE)
image_contour = ktp_image.copy()

# # Initialize variables to store the largest rectangle
max_area = 0
best_rect = None

# Loop through each contour
for contour in contours:
    # Approximate the contour to a polygon
    epsilon = 0.02 * cv2.arcLength(contour, True)
    approx = cv2.approxPolyDP(contour, epsilon, True)

    # Check if the polygon has 4 vertices (rectangle)
    if len(approx) == 4:
        # Calculate the area of the rectangle
        area = cv2.contourArea(approx)

        # Update the largest rectangle if the current one is bigger
        if area > max_area:
            max_area = area
            best_rect = approx

areas = [cv2.contourArea(c) for c in contours]
max_index = np.argmax(areas)

cv2.drawContours(image_contour, contours, -1, (255, 0, 0), 4)
cv2.drawContours(image_contour, [contours[max_index]], -1, (0, 0, 255), 4)
cv2.drawContours(image_contour, [best_rect], -1, (0, 255, 0), 4)

warped = four_point_transform(ktp_image, best_rect.reshape(4, 2))

plt.figure(figsize=(15, 12))

plt.subplot(2, 2, 1)
plt.imshow(cv2.cvtColor(image_contour, cv2.COLOR_BGR2RGB))
plt.title('Image Contour')
plt.axis('off')

# cv2_imshow(image_contour)
plt.subplot(2, 2, 2)
plt.imshow(cv2.cvtColor(warped, cv2.COLOR_BGR2RGB))
plt.title('Image Transformation')
plt.axis('off')
# cv2_imshow(warped)

cropped_images.append(warped)
```

KELOMPOK 5

Hasil Lokalisasi

Image

Image Transformation

Image Contour

Image Transformation

KELOMPOK 5

Segmentasi Image

```
face_detection = []

for img in cropped_images:

  roi_wajah = cascade_wajah.detectMultiScale(img)

  (x, y, w, h) = roi_wajah[0]

  cv2.rectangle(img, (x - 110, y-120), (x + w + 80, y + h+ 100), (0, 255, 0), 2)

  face_det = img.copy()
  face_segment = face_det[y - 120:y + h + 100, x - 110:x + w + 80]

  plt.figure(figsize=(12, 8))

  plt.subplot(2, 2, 1)
  plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
  plt.title('Detected Faces')
  plt.axis('off')

  plt.subplot(2, 2, 2)
  plt.imshow(cv2.cvtColor(face_segment, cv2.COLOR_BGR2RGB))
  plt.title('Face Segment')
  plt.axis('off')

  plt.tight_layout()
  plt.show()

  face_detection.append(face_segment)
```

KELOMPOK 5

Hasil Segmentasi

Image

Detected Faces

Face Segment

Detected Faces

Face Segment

KELOMPOK 5

Face Recognition

Image

Load Model

```
from keras.models import load_model
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Disable scientific notation for clarity
np.set_printoptions(suppress=True)

# Load the model
model = load_model("/content/drive/MyDrive/KTP_KEL5/converted_keras/keras_model.h5", compile=False)

# Load the labels
class_names = open("/content/drive/MyDrive/KTP_KEL5/converted_keras/labels.txt", "r").readlines()
```

KELOMPOK 5

Face Recognition

Image

Face Recognition

```
fig, axes = plt.subplots(1, 5, figsize=(15, 3))

for i, face_segment in enumerate(face_detection):
    # Convert the image to RGB
    image = cv2.cvtColor(face_segment, cv2.COLOR_BGR2RGB)

    # Resize the image
    image = cv2.resize(image, (224, 224), interpolation=cv2.INTER_AREA)

    # Preprocess the image for the model
    image_for_model = np.asarray(image, dtype=np.float32).reshape(1, 224, 224, 3)
    image_for_model = (image_for_model / 127.5) - 1

    # Predict using the model
    prediction = model.predict(image_for_model)

    index = np.argmax(prediction)
    class_name = class_names[index]
    confidence_score = prediction[0][index]

    # Show the image on the i-th subplot with title
    axes[i].imshow(image)
    axes[i].set_title(f'{class_name[2:]}: {confidence_score:.2%}')
    axes[i].axis('off')

    print("Class:", class_name[2:], end="")
    print("Confidence Score:", str(np.round(confidence_score * 100))[:-2], "%\n")

# Tampilkan plot
plt.show()
```

KELOMPOK 5

Hasil Recognition

Image

Rezki
: 99.80%

Sely
: 97.70%

Basith
: 99.95%

Maulana
: 68.79%

Dzaka
: 75.37%

KELOMPOK 5

KESIMPULAN

Untuk menghasilkan pengenalan wajah pada KTP berdasarkan nama pemilik KTP, maka kami melakukan beberapa proses, dimulai dari pengumpulan dataset, preprocessing untuk memperbaiki citra, kemudian dilakukan lokalisasi untuk mengetahui letak objek pada sebuah gambar, lalu segmentasi untuk memisahkan objek dengan background, dan terakhir melakukan pengenalan wajah menggunakan model yang telah dibuat dari website “Teachable Machine”.

Kelompok 5

TERIMA KASIH

APAKAH ADA YANG
INGIN DITANYAKAN?

PCVK