



Desain Sistem Pendeteksi Plat Kendaraan Untuk Pengisian BBM Menggunakan Metode K-Nearest Neighbor (KNN)

Mazlan*, Ewi Ismaredah, Haris Simaremare, Oktaf Brillian Kharisma

Fakultas Sains Dan Teknologi, Teknik Elektro, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru

Panam, Jl. HR. Soebrantas No.Km. 15, RW.15, Simpang Baru, Kota Pekanbaru, Riau, Indonesia

Email: ^{1,*}mazlann1396@email.com, ²ewi.ismaredah@uin-suska.ac.id, ³harrismare@gmail.com, ⁴brilian@uin-suska.ac.id

Email Penulis Korespondensi: mazlann1396@email.com

Submitted: 04/07/2023; Accepted: 22/07/2023; Published: 26/07/2023

Abstrak—Penelitian ini bertujuan untuk merancang dan mengembangkan sebuah sistem pendeteksi plat kendaraan untuk pengisian bahan bakar. Aplikasi yang dibangun menggunakan metode K-Nearest Neighbor (KNN), metode ini digunakan untuk melakukan klasifikasi terhadap plat kendaraan. Penelitian ini akan menggunakan module OpenCV pada bahasa pemrograman python untuk mengenali plat kendaraan dan melakukan tahap pre-processing, segmentasi, ekstraksi fitur, dan klasifikasi data gambar kendaraan. Sistem ini akan memanfaatkan metode K-Nearest Neighbor (KNN) seperti deteksi tepi, transformasi perspektif, dan pengenalan pola untuk mencapai hasil yang optimal. Hasil pengujian menunjukkan bahwa sistem mampu mengenali dengan akurasi tinggi berbagai jenis plat kendaraan dalam kondisi pencahayaan dan sudut pandang yang berbeda, dengan demikian sistem dapat dibuat dan ditentukan untuk masing-masing plat hanya dapat mengisi bahan bakar sejumlah satu kali dalam satu hari. Ketika kendaraan belum pernah mengisi di hari tersebut, maka sistem akan menampilkan lampu hijau yang artinya dapat dilakukan pengisian bahan bakar. Sedangkan bagi kendaraan yang sudah mengisi di hari tersebut, maka sistem akan menampilkan lampu merah yang artinya kendaraan tidak dapat melakukan pengisian sehingga harus menunggu di hari berikutnya setelah sistem di reset. Sistem ini akan dapat secara akurat dan real-time mendeteksi serta mengidentifikasi plat kendaraan yang digunakan dalam proses pengisian bahan bakar. Oleh karena itu, desain sistem ini diharapkan dapat menjadi dasar untuk meningkatkan efisiensi dalam industri pengisian bahan bakar serta memberikan dasar bagi pengembangan penelitian lebih lanjut di bidang ini.

Kata Kunci: Sistem Pendeteksi; Plat Kendaraan; Bahan Bakar; OpenCV; Komputer Vision

Abstract—This study aims to design and develop a vehicle plate detection system for refueling. Applications built using the K-Nearest Neighbor (KNN) method, this method is used to classify vehicle plates. This research will use the OpenCV module in the python programming language to recognize vehicle plates and carry out the pre-processing, segmentation, feature extraction, and classification stages of vehicle Image data. This system will utilize the K-Nearest Neighbor (KNN) method such as edge detection, perspective transformation, and pattern recognition to achieve optimal results. The test results show that the system is able to recognize with high accuracy various types of vehicle license plates under different lighting conditions and viewing angles, thus a system can be created and determined for each license plate to only be refueled once a day. When the vehicle has never been filled on that day, the system will display a green light which means refueling can be done. Whereas for vehicles that have filled in on that day, the system will display a red light which means the vehicle cannot fill up so it must wait the next day after the system is reset. This system will be able to accurately and real-time detect and identify the vehicle plates used in the refueling process. Therefore, the design of this system is expected to be the basis for increasing efficiency in the refueling industry as well as providing a basis for the development of further research in this field.

Keywords: Detection System; Vehicle Registration; Refueling; OpenCV; Computer Vision

1. PENDAHULUAN

Seiring dengan berkembangnya zaman, jumlah penduduk dan industri di Indonesia juga berkembang dan melaju dengan pesat sehingga berdampak langsung terhadap meningkatnya kebutuhan energi nasional [1]. Salah satu elemen yang sangat berpengaruh terhadap perkembangan bisnis bagi sebuah perusahaan adalah Supply Chain Management (SCM). Industri minyak dan gas berperan penting dalam memenuhi kebutuhan energi manusia. Kegiatan hulu sampai hilir menghasilkan produk minyak dan gas yang dapat digunakan dalam berbagai sektor kehidupan. BBM (Bahan Bakar Minyak) merupakan salah satu produk minyak dan gas yang digunakan masyarakat untuk mendukung aktivitas sehari-hari. Di Indonesia, permintaan BBM terus meningkat seiring bertambahnya jumlah kendaraan, infrastruktur jalan, dan aktivitas manusia. Selain untuk kendaraan, BBM juga digunakan sebagai energi penggerak aktivitas di industri.

Pengisian BBM di SPBU secara manual dengan petugas dinilai kurang efektif dan efisien. Sistem tersebut dapat memperlambat waktu pengisian, akibatnya antrean menjadi panjang dan timbul kemacetan. Terbatasnya jumlah petugas membuat jam operasional berkurang, sehingga ada konsumen yang tidak dapat melakukan pengisian BBM karena SPBU sudah tutup [2].

Kurang efektifnya pengisian secara manual berpengaruh terhadap optimalisasi penjualan BBM. Thruput (penjualan BBM) yang tercatat 135 ribu KL per hari yang hanya mengalami kenaikan 10% dibanding pada masa PSBB (Pembatasan Sosial Berskala Besar). Meskipun demikian, prediksi BPH MIGAS (Badan Pengatur Hilir Minyak dan Gas Bumi) pada tahun 2021, kebutuhan konsumen terhadap BBM naik 12,9% atau sekitar 80 juta KL disbanding pada tahun 2020 yang hanya sekitar 70,8 juta KL. Meski kebutuhan BBM naik pada 2021, masih akan lebih rendah dibanding kebutuhan pada tahun 2019 lalu yang mencapai 81,4 juta KL [3].



Solusi permasalahan penjualan BBM adalah menerapkan deteksi plat kendaraan yang merupakan sistem pemantauan dan manajemen kendaraan berbasis plat nomor kendaraan telah berkembang [4][5], identifikasi dan pengenalan plat nomor yang baru baru ini menjadi aplikasi utama dalam bidang pengisian bahan bakar [6]. Berdasarkan wawancara yang dilakukan oleh penulis dengan salah satu karyawan SPBU dikota pekanbaru yang mengatakan bahwa SPBU kerap sekali mengalami kesulitan ketika terjadinya gangguan entry data pada saat pengisian bahan bakar sehingga waktu yang digunakan tidak efisien dikarenakan pada SPBU masih menggunakan manualisasi. Oleh karna itu didesain perancangan pendeteksi plat kendaraan [7]. Tujuan nya adalah untuk mengekstraksi dan mengenali plat nomor dari citra kendaraan untuk mengetahui kendaraan tersebut telah terisi bahan bakar secara otomatis. Telah banyak penelitian mengenai system pengenalan plat nomor. Namun, dalam penelitian ini dikembangkan system pengenalan plat kendaraan setelah melewati batas antrian dan hal ini mampu menjadi terobosan baru dalam ketertiban di SPBU sehingga para pengendara mampu menjadi lebih cepat dan efisien pada waktu. Penelitian ini menggunakan proses ekstraksi pengolahan gambar sehingga metode yang sesuai dengan penelitian ini adalah K-Nearest Neighbor (KNN) yang dilakukan oleh sangsaka wira utama dan Apriani Kusumawardani [8].

K-Nearest Neighbor (KNN) merupakan metode yang biasa digunakan pada klasifikasi data. Algoritma ini digunakan untuk mengklasifikasikan terhadap objek berdasarkan data pembelajaran yang jarak tetangganya paling dekat atau memiliki nilai selisih yang kecil dengan objek tersebut. Salah satu teknik klasifikasi yang paling dasar dan sederhana adalah K-Nearest Neighbor (KNN) [9]. KNN adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Data pembelajaran diproyeksikan ke ruang berdimensi banyak, dimana masing-masing dimensi merepresentasikan fitur dari data Menurut [9] KNN memiliki beberapa keunggulan dan kelemahan, keunggulannya yaitu :1) pelatihan sangat cepat; 2) sederhana dan mudah dipelajari; 3) tahan terhadap data pelatihan yang memiliki derau; dan 4) efektif jika data pelatihan besar. Sedangkan kekurangan dari KNN adalah: 1) nilai k menjadi bias; 2) komputasi yang kompleks; 3) keterbatasan memori; dan 4) mudah tertipu dengan atribut yang tidak relevan. Dengan segala kelebihan dan kekurangan yang dimiliki oleh algoritma KNN, telah banyak permasalahan yang menggunakan algoritma ini sebagai algoritma untuk klasifikasi. Salah satu masalah yang dapat diselesaikan dengan algoritma ini adalah klasifikasi plat kendaraan.

Metode KNN dapat bekerja dengan bantuan OpenCV yang merupakan sebuah library untuk computer vision dan pengolahan gambar yang bersifat open source [10]. Dengan OpenCV akan memudahkan dalam proses pendeteksiannya. Disamping itu efisiensi administrasi juga menjadi pertimbangan penting [11][12]. Dengan sistem pendeteksi plat kendaraan, data plat kendaraan dan riwayat pengisian dapat tercatat dengan lebih akurat. Hal ini dapat membantu pemilik stasiun pengisian bahan bakar dalam meningkatkan efisiensi pengelolaan data [13]. Implementasi sistem ini di stasiun pengisian bahan bakar diharapkan dapat mengurangi risiko penyalahgunaan, memperbaiki manajemen administratif, dan memberikan pengalaman pengisian yang lebih aman dan efisien bagi pengguna kendaraan bermotor. Metode KNN, telah terbukti berhasil dalam berbagai aplikasi pengenalan pola dan deteksi objek [8]. Metode KNN dapat digunakan untuk mendeteksi dan mengenali nomor plat kendaraan dengan tingkat keakuratan yang tinggi [14]. Dengan mengintegrasikan metode KNN pada kamera ALPR, sistem dapat mengambil gambar plat kendaraan saat pengisian bahan bakar dan melakukan proses deteksi kelayakan penerimaan bahan bakar secara otomatis. Metode KNN pada kamera ALPR memungkinkan penggunaan teknologi pengenalan plat nomor kendaraan secara otomatis. Teknologi ini menggunakan kombinasi antara pengolahan gambar, pengenalan pola, dan pembelajaran mesin untuk mendeteksi, mengenali, dan memverifikasi plat nomor kendaraan dengan cepat dan akurat [10]. Sistem ini dapat mengambil gambar plat nomor kendaraan secara real-time dan kemudian menganalisis data tersebut untuk menentukan apakah kendaraan tersebut memenuhi syarat penerimaan bahan bakar. Dengan adanya sistem pendeteksi kelayakan penerimaan bahan bakar menggunakan metode KNN pada kamera ALPR, diharapkan pemerintah dapat meningkatkan efisiensi dalam proses penentuan penerimaan bahan bakar [15].

Berbagai penelitian telah dilakukan sebelumnya terkait dengan penerapan system presensi yang dipadukan dengan beberapa perangkat atau metode baru sehingga mampu merancang suatu system yang memberikan kontribusi tersendiri serta inovasi baru pada setiap system perangkat yang dihasilkan [16][17][18]. Beberapa penelitian telah dilakukan berkaitan dengan sistem pendeteksi kelayakan penerima subsidi bbm diantaranya adalah penelitian yang dilakukan oleh [13] Dengan judul Pengembangan Model Sistem Kelayakan Automatic License Plate Recognition (ALPR) Menggunakan Deep Learning, Penelitian ini akan mengusulkan sebuah metode yang menggabungkan Convolutional Neural Networks (CNN) dan Recurrent Neural Networks (RNN) untuk mengenali dan memverifikasi plat nomor kendaraan dengan lebih baik. Validasi dan evaluasi model akan dilakukan menggunakan metrik kinerja seperti akurasi pengenalan, akurasi verifikasi, dan waktu respons sistem. Penelitian selanjutnya yang dilakukan oleh [15] dengan judul penelitian Penilaian Kelayakan Berbasis Kecerdasan Buatan dari Alpine Ski Resorts, penelitian ini mengeksplorasi penggunaan kecerdasan buatan dalam mengevaluasi kelayakan stasiun ski alpen dari segi keberlanjutan dan efisiensi. Metode yang dikembangkan menggunakan algoritma pembelajaran mesin untuk memodelkan parameter kunci dan memberikan rekomendasi peningkatan [19][20].

Berdasarkan uraian dari permasalahan diatas, penelitian ini berfokus pada pendesainan system pendeteksi plat kendaraan untuk pengisian bahan bakar. System tersebut menggunakan metode KNN dengan bantuan library

OpenCV menggunakan kamera ALPR yang berguna untuk mengambil gambar plat motor kendaraan dan memudahkan kinerja karyawan di SPBU dengan cara system yang otomatis dengan hasil keluaran plat tersebut telah mengisi bahan bakar satu kali atau sudah mengisi yang kedua kalinya. Hasil penelitian yang diharapkan yaitu karena adanya sistem pengisian bahan bakar yang dilakukan satu hari hanya satu kali pengisian bahan bakar, maka sistem akan mendeteksi plat kendaraan yang melakukan pengisian kendaraan. Sehingga ketika terdapat kendaraan yang telah mengisi bahan bakar untuk kedua kalinya dalam jangka waktu satu hari sebelum sistem di reset, maka akan mendapatkan output berupa indikator bahwa kendaraan dilarang mengisi bahan bakar kembali.

2. METODOLOGI PENELITIAN

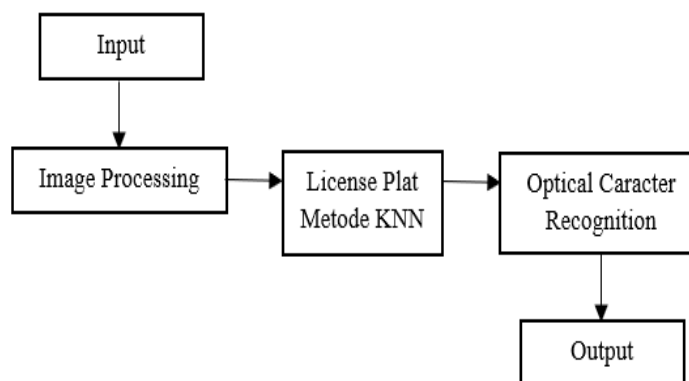
Metode yang digunakan dalam penelitian ini yaitu riset dan pengembangan (Research and Development) [21]. Metode ini adalah metode penelitian yang digunakan untuk menghasilkan system tertentu, dan menguji keefektifan system tersebut, metode ini banyak digunakan dalam mengembangkan system. Alur penelitian yang dilakukan yaitu studi literature, melakukan review paper terkait dengan system yang akan dibuat, melakukan wawancara dengan narasumber untuk menelaah masalah lebih lanjut dan tarik kesimpulan untuk menemukan solusi yang telah teruji dan akan bermanfaat bagi penelitian dimasa mendatang. Langkah kedua membuat rancangan system dengan membuat skema rancangan system keseluruhan, dan dilanjutkan pembuatan diagram rangkaian system. Langkah ketiga adalah mendesain diagram sistem dan langkah keempat adalah analisa dan pembahasan system.

2.1 Metode K-Nearest Neighbor (KNN)

K-Nearest Neighbor (KNN) merupakan metode yang biasa digunakan pada klasifikasi data. Algoritma ini digunakan untuk mengklasifikasikan terhadap objek berdasarkan data pembelajaran yang jarak tetangganya paling dekat atau memiliki nilai selisih yang kecil dengan objek tersebut [22]. KNN merupakan suatu metode yang menggunakan algoritma supervised dengan hasil dari query instance yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN. Tujuan dari algoritma ini ialah mengklasifikasikan objek baru berdasarkan atribut dan training sample. Prinsip umum dari algoritma ini adalah menemukan k data training untuk menentukan K-Nearest Neighbor berdasarkan ukuran jarak. Selanjutnya mayoritas dari k tetangga terdekat akan menjadi dasar untuk memutuskan kategori dari sample berikutnya. Selain itu algoritma ini sendiri sering digunakan untuk klasifikasi pada teknik data mining meskipun dapat digunakan untuk estimasi dan prediksi data.

2.1 Diagram blok sistem

Berikut diagram blok sistem pada penelitian :



Gambar 1. Diagram Blok Sistem

Diagram blok perancangan system ini dapat dilihat pada gambar 1 yang dimulai dengan input sinyal gambar yang masuk ke dalam system, selanjutnya dilakukan Image preprocessing yang proses awal dalam system pendeteksi plat kendaraan untuk menghilangkan noise, meningkatkan kontras dan memperjelas gambar agar pendaftaran plat kendaraan dapat dilakukan dengan lebih mudah. Kemudian dilakukan license plat atau klasifikasi menggunakan metode KNN untuk menganalisis gambar dan mendeteksi wilayah plat kendaraan serta mengisolasi area tersebut dari latar belakang. Selanjutnya proses pengenalan karakter pada wilayah plat kendaraan yang berhasil dilakukan sebelumnya. Dan terakhir hasil pengenalan karakter dari system OCR dapat disimpan kedalam basis data atau ditampilkan sebagai output hasil pengenalan karakter pada layar.

2.2 Perancangan perangkat lunak

Desain perangkat lunak ini sangat penting saat membuat program yang akan dibuat menggunakan python. Program ini diimplementasikan menggunakan perangkat lunak visual studio code [23]. Selanjutnya dilakukan perancangan perangkat lunak berupa flowchart. Berikut flowchart system program python.



Gambar 2. Flowcart Penelitian

Penjelasan Flowchart pada gambar 2 yaitu mulai, lalu mengumpulkan data gambar kendaraan dengan plat nomor yang berbeda selanjutnya melakukan preprocessing gambar dengan mengkonversikan ke skala abu-abu, pengurangan noise, dan peningkatan kontras setelah itu melakukan training model pendeteksi plat kendaraan menggunakan metode KNN dengan bantuan library OpenCV lalu melakukan segmentasi plat kendaraan untuk mendapatkan region of interest (ROI), selanjutnya melakukan pengenalan karakter plat kendaraan didalam ROI, lalu menampilkan hasil deteksi ke pengguna.

3. HASIL DAN PEMBAHASAN

Hasil dan pembahasan akan menjelaskan beberapa pembahasan diantaranya proses program, tahap pengujian sistem dan tahap hasil pengujian sistem. Pada tahap pengujian sistem dilakukan pengujian system yang bertujuan untuk mendapatkan dan menerapkan tahapan untuk mendapatkan hasil pembacaan plat kendaraan dalam bentuk gambar dan karakter. Pada tahap hasil pengujian sistem akan menjelaskan hasil keluaran pada sistem ini sehingga hasil output dapat membantu dan mempermudah untuk meningkatkan efisiensi dalam industri pengisian bahan bakar.

3.1 Proses pemrograman

Pada proses program, dilakukan terlebih dahulu penginputan bentuk gambar plat kendaraan yang kemudian dilakukan path pada gambar untuk dapat diolah dan dideteksi plat kendaraannya, program penginputan dapat dilihat pada program Gambar 3. dibawah ini.

```

ap = argparse.ArgumentParser()
ap.add_argument("-v", "--video", help="Path to video file")
ap.add_argument("-i", "--image", help="Path to the image")
ap.add_argument("-c", "--calibration", help="image or video or camera")
args = vars(ap.parse_args())
img_original_scene = None
loop = None
camera = None

if args.get("video", True):
    camera = cv2.VideoCapture(args["video"])
    if camera is None:
        print("Please check again the path of video or argument !")
        loop = True

elif args.get("image", True):
    img_original_scene = cv2.imread(args["image"])
    if img_original_scene is None:
        print("Please check again the path of image or argument !")
        loop = False
else:
    camera = cv2.VideoCapture(1)
    loop = True
  
```

Gambar 3. Syntax Input Data Sample Plat

Gambar plat motor diolah dan diatur ukuran serta resolusinya sedemikian rupa agar dapat diolah sebagai bentuk gambar citra. Sebelum mengolah gambar, terlebih dahulu dilakukan proses kalibrasi agar hasil yang didapat meminimalisir tingkat error yang terjadi. Program kalibrasi dapat dilihat pada program Gambar 4. berikut.

```

if args.get("calibration", True):
    img_original_scene = cv2.imread(args["calibration"])
    if img_original_scene is None:
        print("Please check again the path of image or argument !")
    img_original_scene = imutils.resize(img_original_scene, width=720)
    cal.calibration(img_original_scene)
    return
  
```

Gambar 4. Syntax Kalibrasi Sample Image

Untuk mendapatkan hasil berupa plat kendaraan, dilakukan proses pencocokan gambar yang telah dimasukkan ke dalam gambar yang terdeteksi. Untuk program segmentasi plat dapat dilihat pada program Gambar 5. dibawah ini.

```
def drawRedRectangleAroundPlate(imgOriginalScene, licPlate):
    # get 4 vertices of rotated rect
    p2fRectPoints = cv2.boxPoints(licPlate.rLocationOfPlateInScene)
    cv2.line(imgOriginalScene, tuple(p2fRectPoints[0]), tuple(
        p2fRectPoints[1]), SCALAR_RED, 2) # draw 4 red lines
    cv2.line(imgOriginalScene, tuple(p2fRectPoints[1]), tuple(
        p2fRectPoints[2]), SCALAR_RED, 2)
    cv2.line(imgOriginalScene, tuple(p2fRectPoints[2]), tuple(
        p2fRectPoints[3]), SCALAR_RED, 2)
    cv2.line(imgOriginalScene, tuple(p2fRectPoints[3]), tuple(
        p2fRectPoints[0]), SCALAR_RED, 2)

def writeLicensePlateCharsOnImage(imgOriginalScene, licPlate):
    # this will be the center of the area the text will be written to
    ptCenterOfTextAreaX = 0
    ptCenterOfTextAreaY = 0
    # this will be the bottom left of the area that the text will be written to
    ptLowerLeftTextOriginX = 0
    ptLowerLeftTextOriginY = 0
    sceneHeight, sceneWidth, sceneNumChannels = imgOriginalScene.shape
    plateHeight, plateWidth, plateNumChannels = licPlate.imgPlate.shape
    intFontFace = cv2.FONT_HERSHEY_SIMPLEX # choose a plain java font
    # base font scale on height of plate area
    fltFontScale = float(plateHeight) / 30.0
    # base font thickness on font scale
    intFontThickness = int(round(fltFontScale * 1.5))
    textSize, baseline = cv2.getTextSize(licPlate.strChars, intFontFace, fltFontScale,
        intFontThickness) # call getTextSize
    # unpack roated rect into center point, width and height, and angle
    (intPlateCenterX, intPlateCenterY, (intPlateWidth, intPlateHeight),
        fltCorrectionAngleInDeg) = licPlate.rLocationOfPlateInScene
    intPlateCenterX = int(intPlateCenterX) # make sure center is an integer
    intPlateCenterY = int(intPlateCenterY)
    # the horizontal location of the text area is the same as the plate
    ptCenterOfTextAreaX = int(intPlateCenterX)
    # if the license plate is in the upper 3/4 of the image
    if intPlateCenterY < (sceneHeight * 0.75):
        ptCenterOfTextAreaY = int(round(intPlateCenterY)) + int(
            round(plateHeight * 1.6)) # write the chars in below the plate
    else: # else if the license plate is in the lower 1/4 of the image
        ptCenterOfTextAreaY = int(round(intPlateCenterY)) - int(
            round(plateHeight * 1.6)) # write the chars in above the plate
    # end if
    textSizeWidth, textSizeHeight = textSize # unpack text size width and height
```

Gambar 5. Syntax Segmentasi Sample Image

Deteksi plat kendaraan harus dilakukan ketika semua proses diatas berhasil dilewati, langkah selanjutnya melakukan deteksi plat yang secara lengkap dapat dilihat pada program Gambar 6. dibawan ini.

```
def detectPlatesInScene(imgOriginalScene):
    listOfPossiblePlates = [] # this will be the return value
    height, width, numChannels = imgOriginalScene.shape
    imgGrayscaleScene = np.zeros((height, width, 1), np.uint8)
    imgThreshScene = np.zeros((height, width, 1), np.uint8)
    imgContours = np.zeros((height, width, 3), np.uint8)
    imgGrayscaleScene, imgThreshScene = Preprocess.preprocess(
        imgOriginalScene) # preprocess to get grayscale and threshold images
    # find all possible chars in the scene,

#####
def extractPlate(imgOriginal, listOfMatchingChars):
    possiblePlate = PossiblePlate.PossiblePlate() # this will be the return value
    listOfMatchingChars.sort()
    key=lambda matchingChar: matchingChar.intCenterX) # sort chars from left to right based on x position
    # calculate the center point of the plate
    fltPlateCenterX = (listOfMatchingChars[0].intCenterX + listOfMatchingChars[
        len(listOfMatchingChars) - 1].intCenterX) / 2.0
    fltPlateCenterY = (listOfMatchingChars[0].intCenterY + listOfMatchingChars[
        len(listOfMatchingChars) - 1].intCenterY) / 2.0
    ptPlateCenter = fltPlateCenterX, fltPlateCenterY
    # calculate plate width and height
    intPlateWidth = int((listOfMatchingChars[len(listOfMatchingChars) - 1].intBoundingRectX + listOfMatchingChars[
        len(listOfMatchingChars) - 1].intBoundingRectX - listOfMatchingChars[
        0].intBoundingRectX) * PLATE_WIDTH_PADDING_FACTOR)
    intTotalOfCharHeights = 0
    for matchingChar in listOfMatchingChars:
        intTotalOfCharHeights = intTotalOfCharHeights + matchingChar.intBoundingRectHeight
    # end for

# this function first finds all contours, then only includes contours that could be chars (without comparison to other
chars yet)
listOfPossibleCharsInScene = findPossibleCharsInScene(imgThreshScene)

# given a list of all possible chars, find groups of matching chars
# in the next steps each group of matching chars will attempt to be recognized as a plate
listOfListsOfMatchingCharsInScene = DetectChars.findListofListsOfMatchingChars(listOfPossibleCharsInScene)
for listOfMatchingChars in listofListsOfMatchingCharsInScene: # for each group of matching chars
    possiblePlate = extractPlate(imgOriginalScene, listOfMatchingChars) # attempt to extract plate
    if possiblePlate.imgPlate is not None: # if plate was found
        listofPossiblePlates.append(possiblePlate) # add to list of possible plates
    # end if
# end for
# print("\n" + str(len(listofPossiblePlates)) + " possible plates found") # 13 with MCLRNF1 image
return listofPossiblePlates
# end function
```

```

fltAverageCharHeight = intTotalOfCharHeights / len(listOfMatchingChars)
intPlateHeight = int(fltAverageCharHeight * PLATE_HEIGHT_PADDING_FACTOR)
# calculate correction angle of plate region
fltOpposite = listOfMatchingChars[len(listOfMatchingChars) - 1].intCenterY - listOfMatchingChars[0].intCenterY
fltHypotenuse = DetectChars.distanceBetweenChars(listOfMatchingChars[0],
                                                  listOfMatchingChars[len(listOfMatchingChars) - 1])
fltCorrectionAngleInRad = math.asin(fltOpposite / fltHypotenuse)
fltCorrectionAngleInDeg = fltCorrectionAngleInRad * (180.0 / math.pi)
# pack plate region center point, width and height, and correction angle into rotated rect member variable of plate
possiblePlate.rectLocationOfPlateInScene = (
tuple(ptPlateCenter), (intPlateWidth, intPlateHeight), fltCorrectionAngleInDeg)
# final steps are to perform the actual rotation
# get the rotation matrix for our calculated correction angle
rotationMatrix = cv2.getRotationMatrix2D(tuple(ptPlateCenter), fltCorrectionAngleInDeg, 1.0)
height, width, numChannels = imgOriginal.shape # unpack original image width and height
imgRotated = cv2.warpAffine(imgOriginal, rotationMatrix, (width, height)) # rotate the entire image
imgCropped = cv2.getRectSubPix(imgRotated, (intPlateWidth, intPlateHeight), tuple(ptPlateCenter))
possiblePlate.imgPlate = imgCropped # copy the cropped plate image into the applicable member variable of the
possible plate
return possiblePlate
# end function

```

Gambar 6. Syntax Deteksi plat

Hasil segmentasi plat kemudian ditulis dalam bentuk text dan juga keluaran gambar yang ditulis dalam program pada Gambar 7.

```

if not loop:
    # write license plate text to std out
    print("license plate read from image = " + licPlate.strChars + "\n")
    # write license plate text on the image
if not loop:
    # re-show scene image
    cv2.imshow("imgOriginalScene", imgOriginalScene)

```

Gambar 7. Syntax Output

3.2 Pengujian sistem

Pengujian diawali dengan menampilkan plat kendaraan bermotor yang nantinya akan dideteksi oleh sistem. Pada gambar 8 dibawah ini menampilkan gambar plat kendaraan bermotor beserta data sistem yang akan diproses.



Gambar 8. Pendeteksi plat sebelum di proses

Preprocessing: Gambar input diproses terlebih dahulu untuk meningkatkan wilayah yang berisi plat nomor. Ini biasanya melibatkan operasi seperti mengubah ukuran, mengonversi ke skala abu-abu, menerapkan filter, dan ambang batas. Untuk plat lokalisasinya program mencari daerah pelat nomor potensial dalam gambar yang telah diproses sebelumnya. Ini dilakukan dengan menggunakan teknik seperti deteksi kontur, deteksi tepi, atau pencocokan template. Selanjutnya dilakukan teknik segmentasi Pelat yang dapat dilihat pada Gambar 9 berikutnya



Gambar 9. Pendeteksi plat saat di proses

Selanjutnya, pada Gambar 9 yaitu segmentasi Pelat yang jika beberapa wilayah pelat nomor potensial ditemukan, program memilih wilayah yang paling mungkin dan memisahkan karakter di dalam pelat. Ini dicapai dengan mengelompokkan karakter berdasarkan sifat spasialnya seperti ukuran, rasio aspek, dan posisi. Selanjutnya pengenalan Karakter, karakter tersegmentasi diteruskan ke sistem pengenalan karakter, yang mengidentifikasi karakter individu berdasarkan fitur visualnya. Ini dapat dilakukan dengan menggunakan teknik pencocokan template. Selanjutnya menampilkan verifikasi yang dapat dilihat pada Gambar 5.

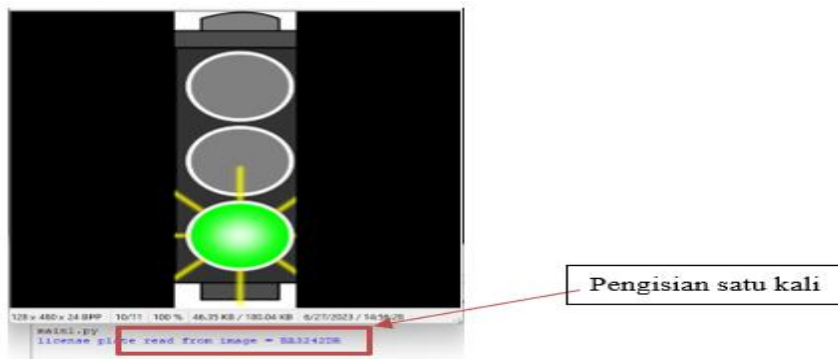


Gambar 10. Pendeteksi plat setelah di proses

Adapun Gambar 10 menampilkan verifikasi untuk meningkatkan akurasi, program dapat melakukan langkah-langkah verifikasi untuk memastikan kebenaran pelat nomor yang terdeteksi. Ini dapat melibatkan pemeriksaan konsistensi karakter yang dikenali pada beberapa frame. Tampilan dan Keluaran program menampilkan gambar asli dengan pelat nomor yang terdeteksi dan karakter atau text yang dikenali disorot.

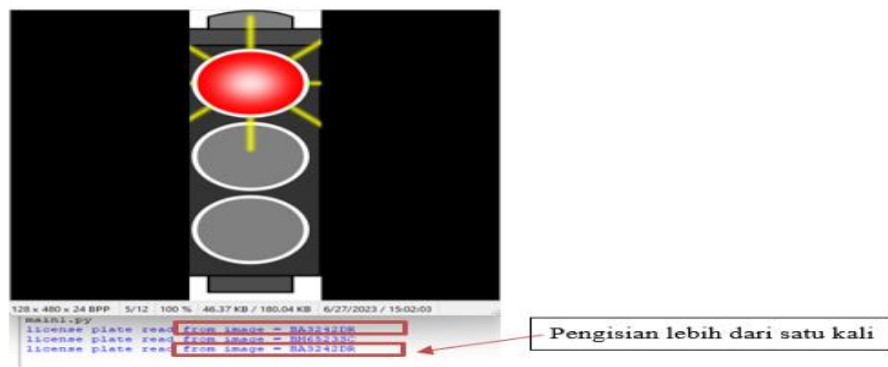
3.3 Hasil pengujian

Hasil pengujian diharapkan dapat memantau kendaraan yang telah mengisi bahan bakar dalam satu hari tidak dapat mengisi bahan bakar untuk kedua kalinya. Sistem ini dibuat dengan mengeluarkan outpun berupa lampu hijau jika kendaraan belum pernah mengisi bahan bakar di hari tersebut, dan output berupa lampu warna merah jika kendaraan tersebut mencoba mengisi bahan bakar untuk kedua kalinya.

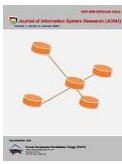


Gambar 11. Pendeteksi kendaraan lolos pengisian

Dapat dilihat pada Gambar 11 menunjukkan bahwa kendaraan tersebut belum pernah mengisi bahan bakar dalam hari yang sama, sehingga akan muncul lampu warna hijau yang menandakan bahwa kendaraan dapat melakukan pengisian bahan bakar. Selanjutnya Gambar 12 menunjukkan bahwa kendararaan mengisi bahan bakar untuk kedua kalinya.



Gambar 12. Pendeteksi kendaraan gagal pengisian



Pada Gambar 12 kendaraan dengan plat BA3242DR melakukan pengisian kedua sebelum sistem di reset di hari berikutnya, hal tersebut akan menampilkan output lampu berwarna merah sehingga pengisian bahan bakar tidak dapat dilakukan dan pengendara dapat melakukan pengisian di hari berikutnya setelah sistem di reset.

4. KESIMPULAN

Dalam kesimpulan, desain sistem pendeteksi plat kendaraan untuk pengisian BBM dengan metode KNN dan library OpenCV merupakan solusi yang efektif dalam meningkatkan efisiensi dan pengawasan dalam proses pengisian bahan bakar. Sistem ini memanfaatkan Metode KNN dan teknologi OpenCV untuk mendeteksi dan mengidentifikasi plat kendaraan secara otomatis, mengurangi ketergantungan pada pengawasan manusia. Penyempurnaan lebih lanjut dapat dilakukan untuk meningkatkan akurasi dan efektivitas sistem ini, serta sistem dapat melakukan pendeteksian bagi kendaraan yang telah mengisi bahan bakar lebih dari satu kali. Pada kendaraan dengan plat BA3242DR melakukan pengisian kedua sebelum sistem di reset di hari berikutnya, hal tersebut akan menampilkan output lampu berwarna merah sehingga pengisian bahan bakar tidak dapat dilakukan dan pengendara dapat melakukan pengisian di hari berikutnya setelah sistem di reset. Sistem ini memberikan manfaat dalam mengurangi risiko penyalahgunaan kendaraan dan memonitor penggunaan kendaraan yang tidak sah. Dengan demikian, desain sistem pendeteksi plat kendaraan ini memiliki potensi besar dan relevansi dalam industri pengisian bahan bakar.

UCAPAN TERIMAKASIH

Puji dan syukur peneliti panjatkan kepada Tuhan Yang Maha Esa atas segala rahmat dan kasih karunia-Nya yang memberikan kesehatan dan kesempatan pada peneliti sehingga paper ini dapat diselesaikan dengan baik. Paper berjudul “Desain Sistem Pendeteksi Plat Kendaraan Untuk Pengisian BBM” disusun untuk memperoleh gelar sarjana teknik Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau. Dalam menyelesaikan paper ini banyak kendala yang dihadapi peneliti dan dapat diselesaikan berkat bimbingan dan dorongan dari berbagai pihak yang akhirnya penulisan ini dapat diselesaikan sebagaimana adanya.

REFERENCES

- [1] I. L. Pratama, “Penerapan self-service berbasis E-card payment dalam mewujudkan digitalisasi penjualan bbm di spbu yang sustainable , efisien , dan profitabilitas,” *INOBS J. Inov. Bisnis dan Manaj. Indones.*, vol. 06, pp. 281–289, 2023.
- [2] H. Muchtar and F. Said, “Sistem Identifikasi Plat Nomor Kendaraan Menggunakan Metode Robert Filter dan Framing Image Berbasis Pengolahan Citra Digital,” *Resist. (elektRonika kEndali Telekomun. tenaga List. kOmputeR)*, vol. 2, no. 2, p. 105, 2019, doi: 10.24853/resistor.2.2.105-112.
- [3] C. Manalu and I. Palandeng, “Analisis Sistem Antrian Sepeda Motor Pada Stasiun Pengisian Bahan Bakar Umum (Spbu) 74.951.02 Malalayang,” *J. EMBA J. Ris. Ekon. Manajemen, Bisnis dan Akunt.*, vol. 7, no. 1, pp. 551–560, 2019.
- [4] A. W. Hamdani and A. Prapanca, “Sistem Deteksi Plat Kendaraan pada Parkiran Rumah Pribadi dengan Metode Background Subtraction dan Optical Character Recognition,” *J. Informatics Comput. Sci.*, vol. 3, no. 03, pp. 250–257, 2021, doi: 10.26740/jinacs.v3n03.p250-257.
- [5] A. Aprilino and A. Al Imam Husni, “Implementasi Algoritma Yolo Dan Tesseract Ocr Pada Sistem Deteksi Plat Nomor Otomatis,” *J. TEKNOINFO*, vol. 16, no. 1, pp. 54–59, 2022.
- [6] R. Humonggio, R. K. Abdullah, and M. Asri, “Pengenalan Plat Nomor Menggunakan Image Processing Pada Perangkat Mikrokontroler,” *J. Teknol. Inf. Indones.*, vol. 4, no. 2, pp. 63–70, 2019, doi: 10.30869/jtii.v4i2.400.
- [7] F. Sunusi, Z. Zainuddin, and S. Sahibu, “Sistem Deteksi Plat Kendaraan Dengan Menggunakan Metode K-Nearest Neighbour (Knn),” *J. Ris. Inform.*, vol. 1, no. 2, pp. 9–14, 2019, doi: 10.34288/jri.v1i2.18.
- [8] S. W. Utama and A. Kusumawardhani, “Aplikasi Pendeteksi Plat Nomor Negara Indonesia Menggunakan OpenCV dan Tesseract OCR pada Android Studio,” no. December, pp. 1–6, 2018.
- [9] A. Budianto, R. Ariyuana, and D. Maryono, “Perbandingan K-Nearest Neighbor (Knn) Dan Support Vector Machine (Svm) Dalam Pengenalan Karakter Plat Kendaraan Bermotor,” *J. Ilm. Pendidik. Tek. dan Kejur.*, vol. 11, no. 1, p. 27, 2019, doi: 10.20961/jiptek.v11i1.18018.
- [10] T. C. A.-S. Zulkhaidi, E. Maria, and Y. Yulianto, “Pengenalan Pola Bentuk Wajah dengan OpenCV,” *J. Rekayasa Teknol. Inf.*, vol. 3, no. 2, p. 181, 2020, doi: 10.30872/jurti.v3i2.4033.
- [11] M. A. Zaimuddin, S. Winardi, S. W. Mudjanarko, and B. Anindito, “Sistem Booking Parkir Mall Dengan Identifikasi Plat Nomor Kendaraan Berbasis Android,” *J. TAM (Technology Accept. Model. Vol.*, vol. 10, 2019.
- [12] B. Santoso and R. P. Kristianto, “Implementasi Penggunaan Opencv Pada Face Recognition Untuk Sistem Presensi Perkuliahan Mahasiswa,” *Sist. J. Sist. Informas.*, vol. 9, no. 2, p. 352, 2020, doi: 10.32520/stmsi.v9i2.822.
- [13] N. Boyko, “Performance Evaluation and Comparison of Software for Face Recognition, based on Dlib and Opencv Library,” 2018 IEEE Second Int. Conf. Data Stream Min. Process., pp. 478–482, 2018.
- [14] M. Zakiyamani, T. I. Cahyani, D. Riana, and S. Hardianti, “Deteksi Dan Pengenalan Plat Karakter Nomor Kendaraan Menggunakan OpenCV Dan Deep Learning Berbasis Python,” *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 5, no. 1, pp. 56–64, 2022, doi: 10.31539/intecom.v5i1.3403.
- [15] W. Sriratana, S. Mukma, N. Tammarugwattana, and K. Sirisantisamrid, “Application of the OpenCV-Python for Personal Identifier Statement,” 2018 Int. Conf. Eng. Appl. Sci. Technol., pp. 1–4, 2018.
- [16] K. Anggara, O. B. Kharisma, A. Wenda, and A. Abdillah, “Smart Early Warning System Untuk Keamanan Sepeda Motor



- Berbasis Prosesor Xtensa Lx6,” *JST (Jurnal Sains dan Teknol.*, vol. 10, no. 2, pp. 135–147, 2021, doi: 10.23887/jstundiksha.v10i2.33425.
- [17] S. Aulia, P. Maria, and R. Ramiati, “Aplikasi Pendeteksi Plat Nomor Kendaraan Berbasis Raspberry Pi Menggunakan Website Untuk Pelanggaran Lalu Lintas,” *Elektron J. Ilm.*, vol. 11, no. 2, pp. 84–89, 2019, doi: 10.30630/eji.11.2.126.
- [18] H. Diwanti, I. S. Sumaryo, and C. Setianingsih, “Real Time Smart CCTV Untuk Mendeteksi Plat Nomor Kendaraan Menggunakan Optical Character Recognition,” *e-Proceeding Eng.*, vol. 6, no. 2, pp. 2–9, 2019.
- [19] M. Rosyadi, R. P. P, and F. T. Industri, “Rancang Bangun Sistem Otomatisasi Gerbang Rumah Dengan Mendeteksi Plat Nomor Kendaraan Berbasis Website,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 6, no. 2, pp. 936–944, 2022.
- [20] M. R. Fauzan and A. P. W. Wibowo, “Pendetections Plat Nomor Kendaraan Menggunakan Algoritma You Only Look Once V3 Dan Tesseract,” *J. Ilm. Teknol. Infomasi Terap.*, vol. 8, no. 1, pp. 57–62, 2021, doi: 10.33197/jitter.vol8.iss1.2021.718.
- [21] H. Sun, M. Fu, A. Abdussalam, Z. Huang, S. Sun, and W. Wang, License plate detection and recognition based on the YOLO detector and CRNN-12, vol. 494. Springer Singapore, 2019. doi: 10.1007/978-981-13-1733-0_9.
- [22] N. Hanum Harani, C. Prianto, and M. Hasanah, “Deteksi Objek Dan Pengenalan Karakter Plat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Python,” *J. Tek. Inform.*, vol. 11, no. 3, pp. 47–53, 2019.
- [23] Y. E. Putra, S. R. Sulistiyanti, and M. Komarudin, “Sistem Akuisisi Data Pemantauan Suhu dan Kadar Keasaman (pH) Lingkungan Perairan dengan Menggunakan Unmanned Surface Vehicle,” *Electr. J. Rekayasa dan Teknol. Elektro*, vol. 12, no. 3, p. 84, 2018, doi: 10.23960/elc.v12n3.2090.