



Deteksi Lubang Jalan Otomatis dengan YOLOv4- Tiny di Google Colab

Pengolahan Citra
Prayitno, S.ST, M.T, Ph.D.

Pengampu : Bapak

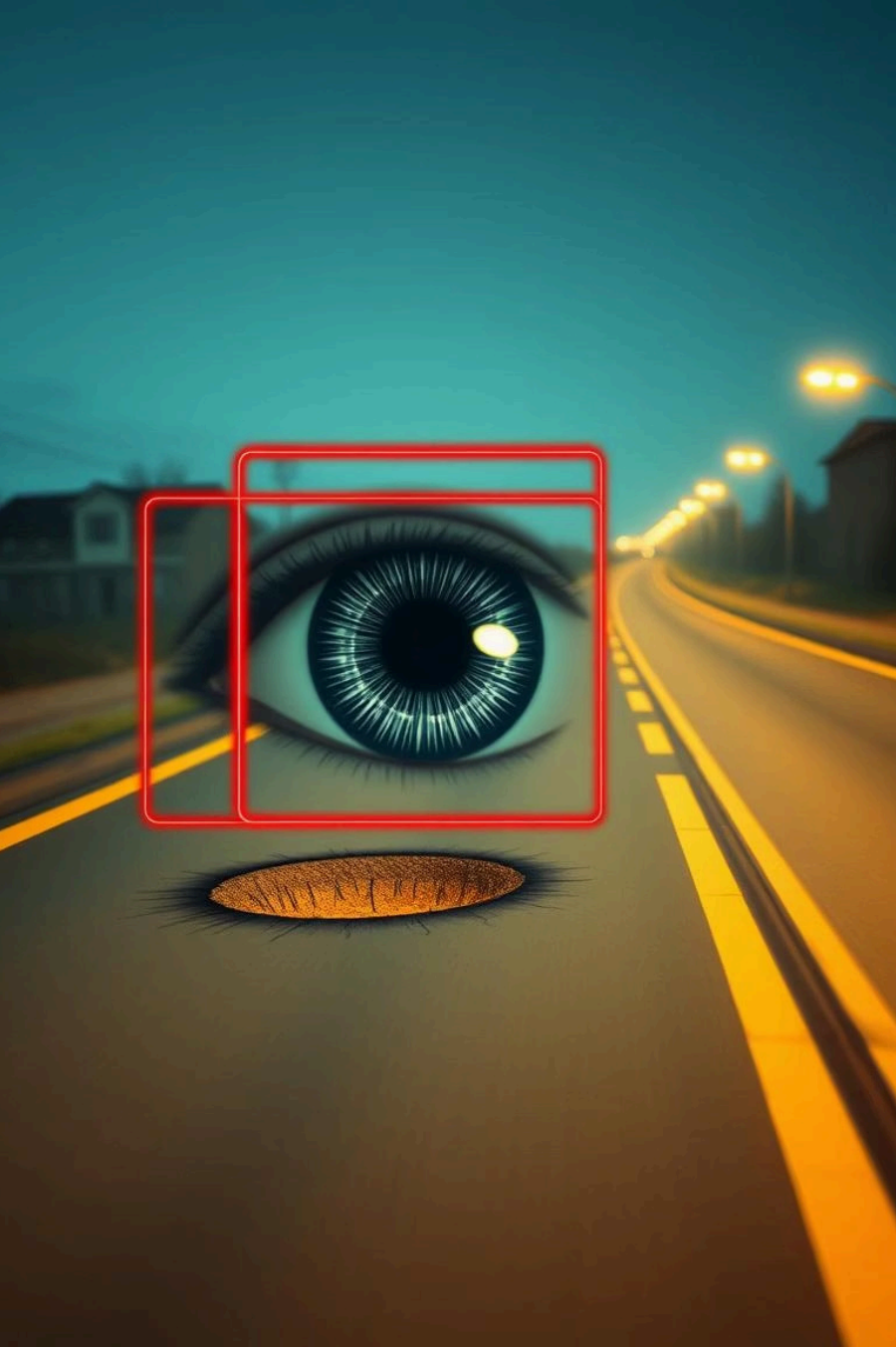


by **Aksobhya S.V**

Mengapa Deteksi Lubang Jalan Penting?

Karena berkaitan dengan keselamatan dan efisiensi. Lubang menjadi penyebab kecelakaan sepeda motor sebanyak 25% (Studi Kementerian PUPR tahun 2022) + merusak kendaraan, Jadi dibutuhkan data kondisi jalan secara real-time agar pemeliharaan dilakukan secara efisien.





Solusi : YOLOv4-Tiny untuk Deteksi Real-time

YOLOv4-Tiny adalah solusi dalam mendeteksi lubang jalan secara real-time. Dengan kecepatan lima kali lebih cepat dari versi biasa dan akurasi mAP (Akurasi Model Deteksi Objek) hingga 80%, model ini bisa memberikan hasil deteksi yang baik tanpa membutuhkan komputer atau server dengan spesifikasi tinggi.

Alat dan Teknologi Utama



Google Colab

Fitur untuk mengunggah video input dan mengunduh hasil video deteksi.



OpenCV

Pemrosesan video dan gambar. Berfungsi membaca, menulis, dan menjalankan



NumPy

Untuk komputasi numerik. Untuk manipulasi array data yang efisien.



Model Pra-terlatih

Menggunakan bobot YOLOv4-Tiny. Dilatih pada dataset lubang jalan spesifik.



Cuplikan Kode (Python)

```
# 1. Install dan import libraries
!pip install opencv-python-headless
import cv2, numpy as np
from google.colab import files

# 2. Upload video
uploaded = files.upload()
video_path = [f for f in uploaded if f.endswith(('.mp4','.avi'))][0]

# 3. Unduh model custom pothole
!wget -q https://raw.githubusercontent.com/akshxyjagtap/Pothole-Detection-System-using-YOLO-Tiny-v4/master/yolov4_tiny.weights
!wget -q https://raw.githubusercontent.com/akshxyjagtap/Pothole-Detection-System-using-YOLO-Tiny-v4/master/yolov4_tiny.cfg
!wget -q https://raw.githubusercontent.com/akshxyjagtap/Pothole-Detection-System-using-YOLO-Tiny-v4/master/obj.names
-O obj.names

# 4. Load model dan label
net = cv2.dnn.readNet("yolov4_tiny.weights", "yolov4_tiny.cfg")
classes = open("obj.names").read().strip().split("\n")
ln = net.getLayerNames()
ols = [ln[i-1] for i in net.getUnconnectedOutLayers().flatten()]

# 5. Setup video IO
cap = cv2.VideoCapture(video_path)
w, h = int(cap.get(3)), int(cap.get(4))
fps = cap.get(cv2.CAP_PROP_FPS) or 25
fourcc = cv2.VideoWriter_fourcc(*'XVID')
out = cv2.VideoWriter("output_pothole.avi", fourcc, fps, (w,h))

# 6. Threshold
CONF, NMS = 0.5, 0.4

# 7. Deteksi per-frame
while True:
    ret, frame = cap.read()
    if not ret: break
    blob = cv2.dnn.blobFromImage(frame, 1/255, (416,416), swapRB=True, crop=False)
    net.setInput(blob)
    layers_out = net.forward(ols)
    boxes,confs,ids = [],[],[]
    for outp in layers_out:
        for det in outp:
            scores = det[5:]
            cid = np.argmax(scores); conf = float(scores[cid])
            if conf>CONF:
                cx,cy,bw,bh = (det[0:4]*[w,h,w,h]).astype(int)
                x,y = cx-bw//2, cy-bh//2
                boxes.append([x,y,bw,bh]); confs.append(conf); ids.append(cid)
    idxs = cv2.dnn.NMSBoxes(boxes,confs,CONF,NMS)
    if len(idxs)>0:
        for i in idxs.flatten():
            x,y,bw,bh = boxes[i]
            cls = classes[ids[i]]
            label = f"{cls}: {confs[i]:.2f}"
            cv2.rectangle(frame,(x,y),(x+bw,y+bh),(0,255,0),2)
            cv2.putText(frame,label,(x,y-10),cv2.FONT_HERSHEY_SIMPLEX,0.6,(0,255,0),2)
    out.write(frame)

cap.release(); out.release()

# 8. Konversi AVI ke MP4
!ffmpeg -loglevel quiet -i output_pothole.avi -vcodec libx264 output_pothole.mp4

# 9. Download hasil
files.download("output_pothole.mp4")
```

Hasil Output





Penjelasan Kode

Kode ini melakukan deteksi lubang jalan dari video menggunakan model YOLOv4-Tiny di Google Colab. Prosesnya mencakup unggah video, unduh model, pemrosesan setiap frame untuk deteksi lubang, penyimpanan hasil ke video baru, konversi ke format MP4, dan akhirnya menyediakan file hasil untuk diunduh.



Terima Kasih