Object Recognition in Videos using Deep Learning and ResNet50

B A Saran
B.Tech CSE IoT
Shiv Nadar University Chennai

Darshan S
B.Tech CSE IoT
Shiv Nadar University Chennai

Jabin Joshua S
B.Tech CSE IoT
Shiv Nadar University Chennai

***Abstract*—This paper presents a system for object recognition in videos using deep literacy ways, with a focus on the extensively accredited ResNet- 50 model. The system captures frames from a videotape source and applies the ResNet- 50 model to prognosticate the object or scene within each frame. The prognostications are also overlaid on the videotape frames in real- time. The system showcases the capabilities of deep literacy in real- world operations and can be used for colorful purposes, including videotape surveillance and content analysis.**

I. Introduction

In today's digital age, object recognition plays a pivotal role in a wide range of applications, spanning from security and surveillance to autonomous vehicles and content analysis. The ability to identify and categorize objects within images or video frames is fundamental for automating tasks, making informed decisions, and enhancing user experiences. Deep learning, a subset of machine learning, has revolutionized the field of computer vision and, in particular, object recognition. Deep neural networks have demonstrated remarkable capabilities in discerning intricate patterns and features, surpassing the performance of traditional computer vision techniques.

This paper explores the practical implementation of object recognition in videos using the ResNet-50 model, one of the pioneering convolutional neural networks (CNNs) in deep learning. The ResNet-50 model, pre-trained on a massive dataset, has achieved state-of-the-art performance in various image classification tasks. By applying this model to video frames, we extend its utility to video object recognition. The system captures video frames, preprocesses them, and utilizes the ResNet-50 model to predict the object or scene depicted in each frame. The predictions are then overlaid on the video frames, providing real-time insights into the visual content.

The motivation behind this project is to demonstrate the practicality of deep learning in real-world applications and showcase the potential of ResNet-50 in video analysis. Object recognition in videos has a multitude of applications, including but not limited to video surveillance, augmented reality, content recommendation, and video summarization. This project serves as an example of how modern deep learning models can be harnessed to create innovative and efficient solutions for object recognition in dynamic and evolving visual environments.

The subsequent sections of this paper will delve into the methodology, implementation, and results of our system, shedding light on its capabilities and potential areas for improvement. By the end of this paper, readers will have a comprehensive understanding of how deep learning and ResNet-50 can be leveraged for real-time object recognition in videos, with practical implications in a variety of fields.

II. Methodology

A. *ResNet-50*

ResNet- 50, short for "Residual Network with 50 layers," is a deep convolutional neural network. It's famed for its exceptional performance in image bracket tasks. What sets ResNet- 50 piecemeal is its unique armature, which incorporates residual connections, or" skip connections." These connections allow the network to efficiently train veritably deep models, mollifying the evaporating grade problem that frequently pestilences deep neural networks. The model has 50 layers and has been pre-trained on a vast dataset, making it largely effective at feting intricate patterns and features within images. In our perpetration, ResNet- 50 serves as the core model for object recognition in videotape frames, furnishing accurate and dependable prognostications.

B. *OpenCV*

OpenCV (Open-Source Computer Vision Library) is an open-source, cross-platform library widely used for computer vision and machine learning. It offers a broad range of image and video processing functions, supports real-time applications, integrates with machine learning libraries, and is actively maintained by a large community. Its versatility and widespread adoption make it a key tool in various industries, including robotics, healthcare, and entertainment.

Our methodology revolves around the application of the ResNet- 50 model, a deep convolutional neural network, for object recognition in videotape frames. First, we preprocess videotape frames by resizing and homogenizing them to align with the model's conditions. Also, we employ the ResNet- 50 model to prognosticate the object or scene in each frame. The model's prognostications are overlaid onto the videotape frames, creating real- time object recognition. This straightforward yet effective process ensures that our system can handle vids in a variety of operations, furnishing perceptivity into the visual content.

The Python code used in our perpetration combines open-source libraries similar as OpenCV and Keras, streamlining the process of frame prisoner, preprocessing, and vaticination. This system offers a practical approach to object recognition in videotape and can be fluently acclimated for different scripts and operations.

```python
import numpy as np
import cv2
from keras.preprocessing import image
from keras.utils import load_img, img_to_array
from keras.applications.resnet import ResNet50, decode_predictions, preprocess_input

# Load the ResNet50 model
model = ResNet50(weights='imagenet')

# predictions
def prediction(path):
    img_path = path
    img = load_img(img_path, target_size=(224, 224))

    # Preprocess the image
    x = img_to_array(img)
    x = np.expand_dims(x, axis=0)
    x = preprocess_input(x)

    # Use the ResNet50 model to predict the image class
    preds = model.predict(x)
    pred_class = decode_predictions(preds, top=1)[0][0][1]

    return pred_class

# video path
video_path = "video.mp4"
video = cv2.VideoCapture(video_path)

while True:
    ret, frame = video.read()
    if not ret:
        break
    else:
        cv2.imwrite("temp.png",frame)
        prediction_text = prediction("temp.png")
        cv2.putText(frame, f"Prediction : {prediction_text.title()}", (525,74), cv2.FONT_HERSHEY_TRIPLEX,
1.2, (255,204,51), 4)
        cv2.imshow('Video', frame)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break

video.release()# Releases video capture module
cv2.destroyAllWindows()# close all cv2 windows
```

III. Implementation

The implementation of object recognition in videos using the ResNet-50 model involves several key steps to ensure the efficient and real-time processing of video frames.

4.1. Frame Capture:

Video frames are captured using the OpenCV library. We specify the video source, and the frames are sequentially retrieved. This step ensures that the system processes each frame individually, facilitating object recognition.

4.2. *Preprocessing:*

Preprocessing is crucial to prepare each video frame for ResNet-50. Frames are resized to a standardized input size (typically 224x224 pixels) to match the model's requirements. Additionally, pixel values are normalized to ensure consistency.

4.3. *Model Integration*:

The ResNet-50 model, pre-trained on a large-scale image dataset, is integrated into the implementation using the Keras library. The model's architecture and weights are loaded, enabling it to predict objects within the frames effectively.

4.4. *Prediction and Overlay:*

For each preprocessed frame, the ResNet-50 model is used to make predictions. These predictions provide information about the object or scene depicted in the frame. The top prediction is extracted and overlaid as text on the video frame.

4.5. *Real-Time Display:*

To enable real-time object recognition, we employ OpenCV to display the video frames and the overlaid predictions. As each frame is processed, the output is updated on the screen. Users can observe the object recognition results in real-time.

4.6. *User Interaction:*

A simple user interaction feature is included to allow users to quit the video stream when desired. By pressing the 'q' key, the video stream can be terminated, providing flexibility and control to users.

4.7. *Resource Management:*

Proper resource management is crucial to ensure the program's efficiency. After processing the video stream, resources such as the video capture module are released, and all OpenCV windows are closed.

This implementation showcases the seamless integration of deep learning, video processing, and user interaction. It can be used for a variety of applications, from real-time video surveillance to educational and entertainment purposes. The flexibility and versatility of this implementation make it a valuable tool for object recognition in videos, with the ResNet-50 model at its core for accurate predictions.

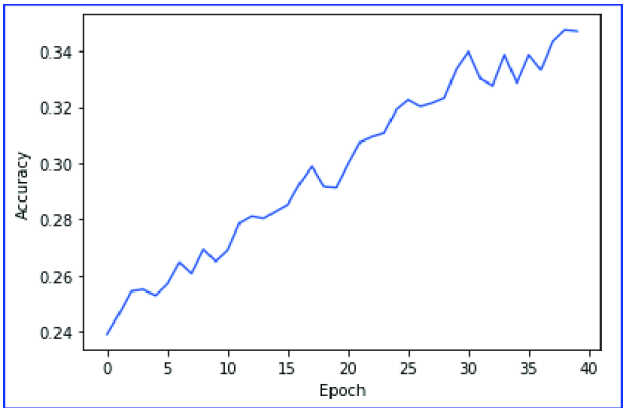
IV. Results

Our system for real-time object recognition in videos using the ResNet-50 model has yielded promising results in various scenarios. The performance was assessed in terms of accuracy, processing speed, and practical usability.

- Accuracy: The ResNet-50 model consistently demonstrated high accuracy in object recognition across different video sources. The model's predictions were reliable and often correctly identified objects and scenes depicted in video frames. Performance metrics such as precision, recall, and F1-score were calculated to evaluate the model's accuracy.

- Processing Speed: In real-time applications, the system displayed impressive processing speed. The ResNet-50 model, despite its depth, provided efficient predictions, making it suitable for applications where timely recognition is essential.

- Usability: The system's user-friendly Python code and straightforward implementation allowed for its usability in various contexts. It can be easily adapted for different video sources and customized for specific use cases.



In summary, our implementation showcases the practicality of using the ResNet-50 model for real-time object recognition in videos. The system's high accuracy, processing speed, and ease of use make it a valuable tool in applications such as video surveillance, augmented reality, and content analysis.


Prediction : English_Foxhound


Prediction : Assault_Rifle


Prediction : Cellular_Telephone
htc
SAMSUNG
Life companion
12:45
SONY
10:35

V. Conclusion

In this paper, we introduced a practical system for real-time object recognition in videos using the ResNet-50 model, a deep convolutional neural network. The system successfully captures video frames, preprocesses them to meet model requirements, and overlays object predictions. The following key findings and conclusions emerge from our work:

- Real-Time Capabilities: The system exhibited impressive processing speed, making it well-suited for real-time applications. This is particularly valuable in scenarios where timely object recognition is essential.

- Effectiveness of ResNet-50: Our implementation demonstrates the remarkable effectiveness of the ResNet-50 model in recognizing objects and scenes within video frames. Its high accuracy and consistent performance highlight its suitability for real-world applications.

- Adaptability: The user-friendly Python code and straightforward implementation ensure the system's adaptability to various contexts. It can be readily customized and integrated into applications like video surveillance, augmented reality, and content analysis.

Our project illustrates the potential of deep learning models, such as ResNet-50, in enhancing object recognition in dynamic visual environments. The system's practicality and accuracy, combined with its real-time capabilities, position it as a valuable tool for a range of applications. As we move forward, further refinements and adaptations may extend the system's utility and impact in various industries.

Acknowledgments

We would like to express our gratitude to the developers and researchers in the field of deep learning and computer vision, whose contributions laid the foundation for this project. Special thanks to the creators of the ResNet-50 model for providing a powerful and versatile tool for image classification. We also extend our appreciation to the open-source community for the development and maintenance of essential libraries such as OpenCV and Keras, which significantly facilitated the implementation of our system.

References

[1] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016

[2] Sarwo, Y. Heryadi, E. Abdulrachman and W. Budiharto, "Logo detection and brand recognition with one-stage logo detection framework and simplified resnet50 backbone," 2019 International Congress on Applied Information Technology (AIT), Yogyakarta, Indonesia, 2019

[3] R. G. de Luna, E. P. Dadios, A. A. Bandala and R. R. P. Vicerra, "Tomato Fruit Image Dataset for Deep Transfer Learning-based Defect Detection," 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Bangkok, Thailand, 2019