Image Captioning and Segmentation

Presented By

Bolle Nihanth Bhargav

Introduction

Image understanding is a fundamental task in computer vision. Two critical sub-problems are:

- **Image Segmentation**: Dividing an image into regions of interest, e.g., background vs. object (dogs, cats, etc.).
- **Image Captioning**: Generating descriptive natural language sentences for images.

Individually, these tasks help in medical imaging, autonomous vehicles, and search engines. By **integrating segmentation with captioning**, the system not only detects *what is present* in an image but also *explains it in natural language*.

This project focuses on developing an integrated pipeline using **Oxford-IIIT Pet Dataset**. The pipeline:

1. Performs **semantic segmentation** of pets (cats and dogs).
2. Generates **captions** describing the scene.
3. Combines results into a single visualization for analysis.

Dataset Used

- **Oxford-IIIT Pet Dataset** (17 breeds of cats and dogs, ~7,000 images).
- Provides **RGB images** + **segmentation masks** with 3 classes:
    - Class 1 → Pet (foreground object).
    - Class 2 → Outline.
    - Class 0 → Background.
- Benefits of this dataset:
    - Balanced across different breeds.
    - High-quality annotated masks.
    - Widely used for benchmarking segmentation tasks.

Methodology

Segmentation Model (U-Net)

- **Architecture:** U-Net (encoder-decoder CNN).
- **Objective:** Pixel-wise classification (foreground, outline, background).
- **Loss Function:** Categorical Cross-Entropy / Dice Loss.
- **Evaluation Metrics:**

- IoU (Intersection over Union): Measures overlap between predicted and ground truth.
- **Dice Coefficient**:Harmonic mean of precision and recall for pixel classification.

Captioning Model:

- **Architecture:** Encoder-Decoder with InceptionV3 + LSTM.
    - Encoder → Pre-trained InceptionV3 extracts features.
    - Decoder → LSTM generates captions word-by-word.
- **Training Objective:** Maximize likelihood of correct captions using teacher forcing.
- **Tokenizer:** Converts words to integer IDs, maintains vocabulary mapping.

Integration Pipeline:

- Step 1: Preprocess input image (resize + normalize).
- Step 2: Predict segmentation mask using U-Net.
- Step 3: Extract CNN features + generate caption using trained LSTM.
- Step 4: Display results (original image, ground truth mask, predicted mask, caption).

Results

Segmentation Performance:

- **Average IoU:** ~0.70
- **Average Dice Coefficient:** ~0.82
- **Observations:**
    - Good performance on clear foreground objects.
    - Some difficulty with complex backgrounds and overlapping pets.

Captioning Performance

- **Example outputs:**
    - **Input:** Dog in grass → Caption: *“a brown dog is running in a field”*.
    - **Input:** Cat on sofa → Caption: *“a white cat is sitting on a couch”*.
- **Issues:**
    - Captions sometimes generic (“a man in a black shirt…” due to pretrained model bias).
    - Vocabulary size and dataset size limited expressive ability.

Integrated Output

The final visualization shows:

1. Original Image.
2. Ground Truth Mask.

3. Predicted Segmentation Mask.
4. Generated Caption.

This integration proves the ability of the system to both **understand structure** (segmentation) and **describe content** (captioning

Applications

- **Healthcare:** Identifying and describing anomalies in medical scans.
- **Autonomous Vehicles:** Detecting and describing objects in real-time.
- **Search Engines:** Automatic tagging and captioning for large image datasets.
- **Accessibility:** Helping visually impaired users understand images through spoken captions.

Conclusion & Future Work

This project successfully built an **end-to-end integrated pipeline** for segmentation + captioning.

- **Achievements:**
  - Developed U-Net segmentation achieving IoU ~0.70.
  - Implemented captioning model with reasonable descriptive accuracy.
  - Integrated both systems into a final visualization pipeline.
- **Limitations:**
  - Captions biased due to pretrained models (ImageNet captions).
  - Segmentation errors in complex backgrounds.
  - No large-scale training due to hardware constraints.
- **Future Work:**
  - Fine-tuning captioning model on domain-specific data.
  - Improving segmentation using deeper architectures (e.g., Mask R-CNN, DeepLabV3+).
  - Deploying as a real-time web/mobile app using Streamlit or Flask.