INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT (IJSREM)
VOLUME: 05 ISSUE: 06 | JUNE - 2021
ISSN: 2582-3930

Windows Software for Image labelling for Creating Database for Insight into Images

Dr. Josephine Prem Kumar
Professor,
Dept. of Computer Science and Engg.,
Cambridge Institute of Technology,
Bangalore, India.

Anshika Dixit
Dept. of Computer Science and Engg.,
Cambridge Institute of Technology,
Bangalore, India.

Ateev Gupta
Dept. of Computer Science and Engg.,
Cambridge Institute of Technology,
Bangalore, India.

Harsh Raj
Dept. of Computer Science and Engg.,
Cambridge Institute of Technology,
Bangalore, India.

The goal of this project is to develop a tool that will semantically organize the descriptions of the object to be recognized and extract information from 'unlabeled urban scene images.

The program will draw polygons enclosing the different objects in an image and return the date, label and the coordinates of the polygon in the form of overlay drawn over the image as a Json File. This tool can be used to understand and extract information for autonomous vehicle monitoring, urban studies and urban modelling, civil and urban engineering.

This work will offer urban modellers a realistic platform for documenting city dynamics and tackling various urban issues. Hence, a computer vision tool will be developed for multipurpose urban modelling that can be used for data visualization or plugin any statistical, deep learning model to carry-on further analysis or prediction on acquired data.

I. INTRODUCTION

With the emergence of Internet of Things and cyber infrastructure, urban areas are passing through a rapid development towards digital cities and even smart cities. With the growth of the field of deep learning and computer vision, understanding cities through the eyes of a computer opens the door for analysing various missing attributes of city dynamics. The complex characteristics of the complex urban objects might be useful for landscape planning, land management, and traffic monitoring in the era of digital economy; these are to be effectively and efficiently recognized. The cost of data preparation for the deep learning-enhanced object recognition is expensive. The amount of object type that a CNN specially designed for object recognition can recognize seems limited and heavily relies on human's labeling. Creating our own dataset was the only way to conduct this deep learning model.

II. ARCHITECTURE

Fig. 1 shows the Architectural Diagram.

Fig. 1 Architectural Diagram

A. Image Processing

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which the input is an image and output may be image or characteristics/features associated with that image. Nowadays, image processing is among rapidly growing technologies. It forms core research area within engineering and computer science disciplines too.

Image processing basically includes the following three steps:

- Importing the image via image acquisition tools;
- Analyzing and manipulating the image;

- Output in which result can be altered image or report that is based on image analysis.

B. Image Annotation

Image annotation is defined as the task of annotating an image with labels, typically involving human-powered work and in some cases, computer-assisted help. Labels are predetermined by a machine learning engineer and are chosen to give the computer vision model information about what is shown in the image. The process of labeling images also helps machine learning engineers hone in on important factors that determine the overall precision and accuracy of their model. Example considerations include possible naming and categorization issues, how to represent occluded objects, how to deal with parts of the image that are unrecognizable, etc.

C. Storing data in Json

JSON is a file format that's used to store and interchange data. Data is stored in a set of key-value pairs. This data is human readable, which makes JSON perfect for manual editing.

II. III. PROPOSED MODEL

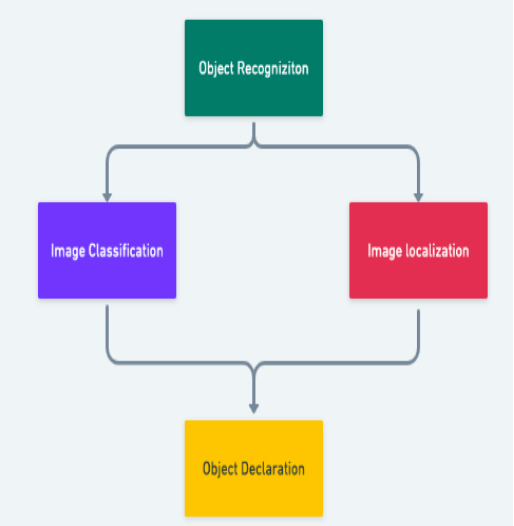


Fig. 2 Proposed Model

Fig. 2 describes broadly the proposed model.

A. Object Recognition

Object recognition is a computer vision technique for identifying objects in images or videos. Object recognition is a key output of deep learning and machine learning algorithms. When humans look at a photograph or watch a video, we can readily spot people, objects, scenes, and visual details. The goal of object recognition is to teach a computer to do what comes naturally to humans: to gain a level of understanding of what an image contains.

B. Image Classification

Image Classification is a fundamental task that attempts to comprehend an entire image as a whole. The goal is to classify the image by assigning it to a specific label. Typically, Image Classification refers to images in which only one object appears and is analyzed. In contrast, object detection involves both classification and localization tasks, and is used to analyze more realistic cases in which multiple objects may exist in an image.

C. Image Localization

Object Localization is the task of locating an instance of a particular object category in an image, typically by specifying a tightly cropped bounding box centered on the instance. An object proposal specifies a candidate bounding box, and an object proposal is said to be a correct localization if it sufficiently overlaps a human-labeled "ground-truth" bounding box for the given object. In the literature, the "Object Localization" task is to locate one instance of an object category, whereas "object detection" focuses on locating all instances of a category in a given image.

D. Object Declaration

Object Declaration is the goal of uniquely labelling each pixel as being part of a connected object. It is related to the tasks of segmentation and thresholding, but when binary images are being discussed the task is usually referred to as labelling.

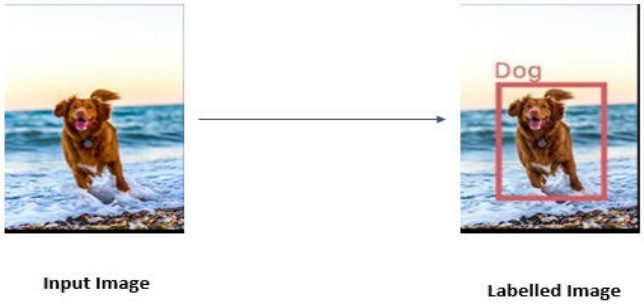
III. IV. IMPLEMENTATION WITH PYTHON

The GUI is developed on **TKinter**. The user selects the image using the GUI from his local storage of the system. The tool allows the user to adjust the image properties such as the specific region. It even allows the user to zoom in and out of the picture. The GUI tool also allows the user to orientate the selected image and choose the exact Region of Interest (implemented using **OpenCV**). Once the Region of Interest is selected the software allows the user to label the ROI. The software snips out the ROI along with the label and other attributes such as the dimension of the image, and the particular class the object inside the image belongs to.

The JSON file is generated from the image, which contains all the parameters of the image such as the height and the dimension of the image in an object notation format; the objects are the parameters of the image selected.

IV. V. RESULTS

Fig. 3 shows the input image, then input image is labelled. The result of the Windows software is a JSON file, as shown in Fig. 4 containing all the properties of the image.



As shown Fig. 4 the JSON file has the various labelled object with their dimensions.


```
{
    "imgHeight": 1920,
    "imgWidth": 2560,

    "objects": [
  {
            "date": "13-April-2021 11:02:38",
            "deleted": 0,
            "draw": true,
            "id": 0,
            "label": "1 - dog",
      "polygon": [
                [
                    675.675789767787,
                    2559.0
                ],
                [
                    2536.296296296296,
                    2559.0
                ],
                [
                    675.675789767787,
                    244.93827160493825
                ],
                [
                    2536.296296296296,
                    244.93827160493825
                ],

            ]
  }
```


Fig. 4 Json File for the object 'dog'

Fig. 3 Example Input Image

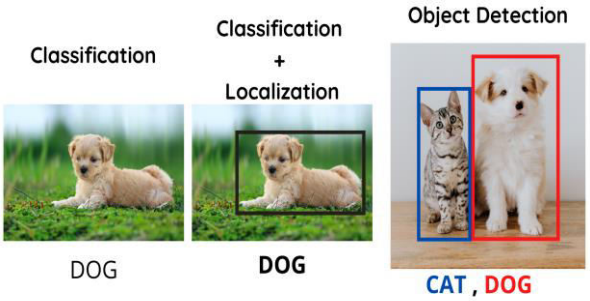


Fig. 5

Fig. 5 depicts the following, which is implemented on our project:

1. Image Classification
2. Image Localization
3. Object Detection

V. VI. CONCLUSION

A novel approach is proposed in this paper to automatically generate pixel level object labellings. In this we will take the coordinates of each and every pixel of the image based on the mouse clicks of the user, and read the coordinates and then store it in the JSON file.

JSON file is one which is leading format for storing data, and as compared to other formats it requires very less space because there are no schemas involved. This format can send data between browser and server, since it is platform independent and its simple design and flexibility makes it easy to read and understand. Iin most cases it's easy to manipulate in the programming language of our choice.

Further this data can be used to precisely train the CNN model.

VI. VII. REFERENCES

[1] X. He and S. Gould. An exemplar-based CRF for multiinstance object segmentation. In CVPR, 2014.

[2] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? Trans. PAMI, 38(4):814–830, 2015.

[3] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. OverFeat: Integrated recognition, localization and detection using convolutional networks. In ICLR, 2014.

[4] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, Y. Zhao, and S. Yan. STC: A simple to complex framework for weakly-supervised semantic segmentation.

[5] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015.

[6] J. Yao, S. Fidler, and R. Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In CVPR, 2012.

[7] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A.W. M. Smeulders. Selective search for object recognition. International journal of computer vision, 104(2):154–171 2013