IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, May 2026
Impact Factor: 8.2

IndiaAnnotate: A Curated Marketplace for Localized AI Datasets

Proff Renu Kachoria, Madhav M. Baviskar, Aditya Ravi Shetty, Meenal Shendre, Sujal Bhelke, Harsh Chendwankar
Department of Engineering, Sciences and Humanities (DESH)
Vishwakarma Institute of Technology, Pune, Maharashtra, India

Abstract: *Accurate annotated datasets represent the key to high-performance computer vision model development. The manual, extremely time consuming, expensive and prone to errors approach to building annotated datasets is commonly known as the "Annotation Bottleneck". The main objective of this paper is to present a comprehensive "Automated Dataset Annotation and Validation System". This system uses state-of-the-art YOLOv8 object detection architecture to generate bounding box annotations automatically in standard COCO format. The system includes a robust, schema-based validation system, which is made accessible through a Flask-based API, and a modern front-end. The proposed system can take a directory with raw images, make inferences on them in real-time, construct structurally valid COCO JSON files, and validates them based on several quality metrics (label balance distribution, bounding box coverage ratios and schema conformance). In comparison with the manual annotation process the developed system leads to a more than 90%time reduction and a high consistency in terms of output data structure, thereby offering a general, re-usable solution for quickly bootstrapping computer vision datasets in various application fields.*

Keywords: Automated Dataset Annotation, YOLOv8, COCO Format, Object Detection, Dataset Validation, Computer Vision, Flask API

I. INTRODUCTION

Deep learning has shown great success in computer vision with a wide variety of applications ranging from intelligent surveillance, medical imaging and autonomous driving to robotics. These applications strongly rely on large-scale, well-annotated datasets for training and evaluating the deep neural networks, but with deep networks increasing in depth and width, the demand for annotation services is also increasing drastically.

Even with the tremendous progress made in deep network architectures, annotated data generation remains a labor-intensive and critical part of the machine learning lifecycle. Manual annotation of data in tools like CVAT, LabelMe and LabelImg involves the laborious process of drawing bounding boxes and segmentation masks on images. Though these tools offer a flexible interface, manual labeling can be error prone, inefficient, and can be highly time-consuming, potentially taking up to several hundreds of hours for a dataset of a few thousand images.

The advancements in object detection models have facilitated partially automating the process of data annotation, as state-of-the-art detection architectures such as YOLO are proven to achieve accurate real-time detection performance suitable for generating initial annotations on unlabeled data. However, current approaches typically only focus on detection outputs, lacking a standardized output format or a way to assess the quality of the generated dataset. There is a clear need for a system that not only automatically generates annotations for an unlabeled dataset but also ensures that they adhere to a particular structure and satisfy a set of predefined quality requirements.

To bridge this gap, this paper introduces IndiaAnnotate, an end-to-end automated system that takes an image dataset as input and produces a training-ready computer vision dataset with automatic object annotations. The system uses a YOLOv8 based inference pipeline to detect objects and generate bounding boxes for input images, a transformation pipeline that converts the detection output into the standard COCO dataset format, and a validation engine to inspect

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, May 2026 Impact Factor: 8.2

the structure and content of the generated dataset using different statistical metrics such as structural compliance, annotation coverage, class balance and so on. A simple dashboard is provided to quickly view the dataset statistics and infer the reliability of the generated dataset.

The research questions addressed in this work are:

• **RQ1:** To what extent can automated object detection models be used to bootstrap dataset annotation for computer vision datasets?

• **RQ2:** Can automated validation metrics effectively evaluate the structural and statistical quality of generated datasets?

• **RQ3:** How much efficiency improvement can an automated annotation pipeline provide compared to traditional manual labeling workflows?

The primary contributions in this work based on the research questions can be summarized as follows:

• An automated annotation pipeline that leverages YOLOv8 for object detection and annotation, and produces dataset in COCO format.

• An automated validation engine to perform checks for structural consistency and statistical validity of the generated dataset.

• An end-to-end integrated system combining automated annotation, data formatting and validation, supported by a lightweight analytical dashboard.

• An experimental validation of the system to quantify the efficiency gains over manual labeling for large datasets while maintaining data structure integrity.

Through these contributions, the proposed framework seeks to facilitate data-centric development for computer vision applications.

II. LITERATURE REVIEW

It's clear that High quality, well-annotated datasets are of paramount importance when it comes to training current computer vision models, and as such significant effort has been made to simplify the process of labeling datasets.

Initially, manual labeling tools were used to annotate datasets. LabelMe is one example of such a tool developed by Russell et al. [7]. This is a web-based system designed for collaborative annotation, allowing users to hand label images via bounding boxes or polygons, providing a foundation for structured image databases. While flexible and highly precise, manual annotation tasks can be time consuming, and difficult to scale for large image sets.

To minimize reliance on manual labor, object detection techniques have been integrated into the labeling process. YOLO (You Only Look Once) [1] a popular object detection algorithm introduced by Redmon et al. Enables accurate detection of bounding boxes and class probabilities within a single feed-forward operation of a deep learning network. Recent developments in the YOLO framework have only increased their capabilities, with current models delivering superior accuracy and speed. Object detection models are often used to create initial "rough" annotations, which can then be further processed, or may serve as an initial step in more labor-intensive annotation workflows.

Furthermore, standard object detection formats have been proposed to allow easier transfer of information between different machine learning platforms. The COCO (Common Objects in Context) dataset format introduced by Lin et al. [2] has become the industry standard for object detection datasets. The structure of COCO stores a comprehensive collection of image, object, and annotation metadata in a structured JSON format that can be easily processed by many modern computer vision libraries. Unfortunately, translating the output of object detection algorithms into COCO format usually involves further pre-processing and validation.

Although, individual components for annotation, object detection, and standardized dataset formatting now exist, they have generally been used independently in traditional dataset preparation processes. Many approaches either still require fully manual annotation, or utilize object detection models without a facility for converting detection outputs into a standard dataset format. Most available tools used for validating datasets only check structural consistency, and

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, May 2026
Impact Factor: 8.2

do not necessarily take into account potential statistical anomalies, or deficiencies in the coverage or accuracy of the annotations.

The clear requirement is thus, a unified system capable of automating annotation through object detection and translating it into a standard dataset format along with inherent means to evaluate the statistical and intrinsic quality of the data itself. Our system fulfills this requirement by enabling full automated annotation of datasets, generating validated COCO formatted data while simultaneously providing rigorous evaluation of annotation quality and dataset statistics.

System Architecture

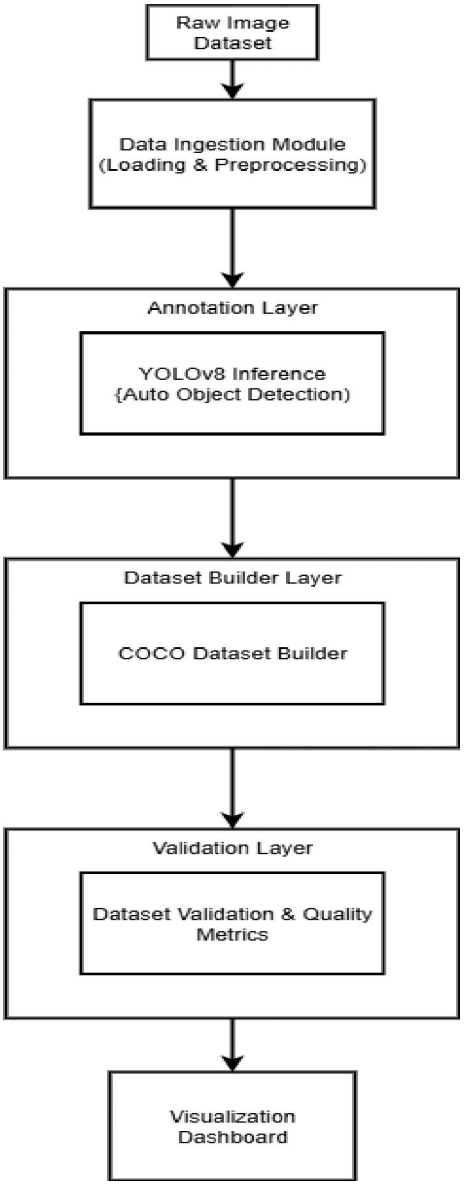

Fig. 1

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, May 2026 Impact Factor: 8.2

The IndiaAnnotate framework is proposed as an end-to-end pipeline designed for automatic conversion of image collections into well-validated, training-ready datasets for computer vision applications. The framework is integrated with automatic object detection, standardization of the dataset format, and validation of the dataset quality into one architecture. The system, as described in Fig. 1, takes images as input, and they flow through connected modules which are used for the purpose of generating annotation, constructing, and validating quality of the dataset.

The architecture includes 5 components: **data ingestion module, YOLO inference engine, COCO dataset builder, dataset validation engine, and a visualization dashboard.** All these components of the system are used for the construction of the automated process of converting unstructured images into the structured dataset to be used for training computer vision models.

Data Ingestion Module

This is the entrance point to the framework where a directory of raw images is ingested. Supported image formats like JPG, PNG, JPEG are detected by scanning the entire input directory. Initial preprocessing on images like resizing and normalization is done so that it is compatible with the input requirements of the detection model. Crucial metadata such as the filename, actual image dimensions are also recorded in order to be used in later stage of dataset construction.

YOLO Inference Engine

The automated object detection is done by YOLO inference engine. A pre-trained YOLOv8 model is employed here to detect the objects present in the image and make bounding box predictions. For every image, the system extracts a set of detections comprising of predicted class of object, confidence score, and bounding box coordinates. These detections can serve as the raw annotation to the dataset. An efficient object detection model such as YOLO reduces the time for manual annotation to a greater extent and maintain annotation consistency.

COCO Dataset Module

The generated detection by the YOLO model is processed by the COCO dataset builder module, where the raw output of detections are converted into COCO dataset format. There are three main parts of the COCO dataset, namely images, annotations and categories. The bounding box coordinates predicted by the YOLO model is converted into absolute pixel values from normalized values (as expected in COCO format). Unique IDs are generated for images and annotations in the dataset.

Dataset Validation Engine

To make sure that the resulting dataset is reliable, a dataset validation engine is incorporated. The validation process includes schema compliance validation where it is checked if all necessary fields for a dataset file are present and they are in proper structure. Furthermore, the statistics about the annotation are also computed such as annotation coverage for the image and class distribution balance. These two are the essential indicators of the overall quality and consistency of the dataset generated.

Visualization Dashboard

A lightweight visualization dashboard is present in the framework to display the information about the created dataset. Some useful information like number of annotated images, class distribution and validation metrics of the dataset can be displayed in the dashboard, helping the users understand the quality and validity of the dataset being used.

In summary, this framework proposes a system that will serve as a single pipeline from raw images to well-structured and verified dataset for computer vision training. With combination of detection, dataset normalization and verification in one pipeline, it will significantly improve the efficiency and speed of dataset preparation for computer vision application.

Methodology/Experimental

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, May 2026 Impact Factor: 8.2

The IndiaAnnotate framework is intended to enable the automatic transformation of an unannotated image collection into a machine learning training-ready structured dataset. This work describes a technique for automatic annotation of images, translation to a standard dataset format, and validation of the resultant dataset's quality. Each phase is completed in sequence from ingestion of images to validation of quality.

Input raw images are first acquired by the application. The application scans a designated directory and detects supported image formats including JPG, PNG and JPEG. Images are then preprocessed before they can be used by an object detection model. This involves resizing of images to match input size expectations for the YOLOv8 model as well as normalization of pixel values to required ranges. This ensures consistent inputs to and predictions from the detection model. Image dimensions (height, width), as well as the file identifier (which will be used in the constructed dataset), are saved at this stage.

After preprocessing, automatic annotation of images with objects of interest is performed by use of the YOLOv8 object detection model. The YOLO (You Only Look Once) object detection architecture was chosen due to its effective compromise of both computational efficiency and accuracy. This model, unlike prior multi-stage detectors (region proposals, feature extraction and classification), executes object detection in a single pass over the network and is hence particularly well-suited for bulk processing and automatic dataset construction from an image collection.

Given an input image I, the object detection model generates a list of detections that contain objects within it. The set of detections can be stated as:

$$D = \{d_1, d_2, \ldots, d_n\}$$

where, each element $\boldsymbol{d_i}$ represents an object instance that has been detected in the image. Each detection contains the object class predicted as well as the prediction confidence and dimensions of a bounding box around the detected object:

$$d_i = (c_i, s_i, x_c, y_c, w, h)$$

where:

$\boldsymbol{c_i}$ is the object class predicted

$\boldsymbol{s_i}$ is the confidence of the prediction

$\boldsymbol{x_c}, \boldsymbol{y_c}$ are the center coordinates of the bounding box

$\boldsymbol{w}$ and $\boldsymbol{h}$ are the width and height of the bounding box

These coordinates are expressed in **normalized form**, meaning they are defined relative to the dimensions of the image rather than in absolute pixel units.

However, most computer vision datasets (including the common COCO dataset) use absolute pixel coordinates for bounding boxes, hence the YOLO coordinates need to be converted before the dataset can be generated.

Given $\boldsymbol{W_{img}}$ and $\boldsymbol{H_{img}}$ are the width and height of the image respectively, the conversion between YOLO and COCO bounding box format follows the rules below:

$$x_{min} = (x_c - \frac{w}{2}) \times W_{img}$$
$$y_{min} = (y_c - \frac{h}{2}) \times H_{img}$$
$$w_{abs} = w \times W_{img}$$
$$h_{abs} = h \times H_{img}$$

This transforms the center coordinates used by YOLO into the top-left coordinates and pixel values used in COCO format: $\boldsymbol{x_{min}}$ and $\boldsymbol{y_{min}}$ represent the top-left corner of the bounding box, while $\boldsymbol{w_{abs}}$ and $\boldsymbol{h_{abs}}$ represent the width and height in pixels, respectively.

The area of each detected object is then calculated using its absolute pixel dimensions:

$$Area = h_{abs} \times w_{abs}$$

This is required by COCO, but also serves to show the scale of object instances in the image collection.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, May 2026 Impact Factor: 8.2

Using the converted absolute pixel bounding box information and the stored metadata, the dataset builder now constructs the final dataset structure according to the COCO annotation specification. The resultant dataset comprises three elements:

Images, which store metadata about each image
Annotations, which store the bounding boxes and object labels
Categories, which define the list of object classes

Each annotation record includes the file identifier and a category identifier in order to link annotations back to their image and category respectively.
A final step involves evaluating the generated dataset to ensure it is suitable for training a machine learning model. Datasets generated automatically through a computational process may still be inappropriate for use with machine learning if the generation mechanism produces datasets with undesirable statistical properties. This work introduces two main components to the dataset validation stage: structural validity and statistical validity. Schema validation checks the conformity to COCO structure whereas class imbalance and coverage statistics give an insight into overall quality.
Overall quality of the dataset is presented using the **dataset quality score Q:**

$$Q = w_1(S_{schema}) + w_2(R_{coverage}) + w_3(1 - E_{imbalance})$$

where;
S_{schema} is the validation score of dataset structure (schema check)
$R_{coverage}$ is the proportion of dataset images containing annotations
$E_{imbalance}$ is the measure of imbalance between object classes
w_1, w_2, w_3 are the weight coefficients of the validation scores
A higher quality score indicates a dataset is more suitable for training a machine learning model.
By integrating automatic annotation with dataset construction and quality validation in one sequence, we have created a useful framework for quickly generating high quality annotated image datasets from raw image collections.

III. RESULTS AND DISCUSSIONS

The presented IndiaAnnotate framework is validated on the basis of performance for automated dataset creation, as well as evaluating the generated dataset quality on the base set of experiment. Three components are considered during the discussion of performance, of which, 1) the performance of automated annotation and 2) the dataset quality produced, and 3) performance with the traditional methods. The experiment is to highlight how automation within the proposed IndiaAnnotate pipeline for detecting objects, formatting the datasets, and validating the quality can help ease out and fasten the overall dataset creation procedure in computer vision tasks.

Experimental Procedure

The validation of the proposed framework is performed on sample image datasets for several common object categories that are expected to appear in many datasets. The system configuration is provided with Intel i7 CPU and NVIDIA GTX series GPU, so that inference by YOLOv8 detection model can be achieved more effectively.
By supplying raw image datasets, the IndiaAnnotate pipeline will produce object detections and convert them into the COCO-based annotations. The generated dataset will then be evaluated by some structural and statistical methods.

The experiment will examine two factors: ***the quality of dataset annotation structure and the characteristics of dataset distribution statistics.***

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, May 2026
Impact Factor: 8.2

Performance of Automatic Annotation

Reducing human time and labor to build an annotated dataset is one of the most significant advantages of the proposed framework. In the manual annotation scenario, where each annotator uses a specific tool to draw bounding boxes around each object, a huge amount of time will be taken for the large datasets. Thus, by using of object detection model such as YOLOv8 the proposed system has successfully taken off much of human workload and accelerated the overall dataset building procedures in computer vision, especially with respect to large datasets. Additionally, it offers greater consistency compared to the human annotations, where a subject has subjective interpretations and may label objects differently.

Quality of Dataset Evaluation

While automatic annotation is an important goal of the proposed IndiaAnnotate framework, evaluating the quality of the generated dataset is also crucial since incorrect data structures or heavy class imbalance may cause undesirable performance of the machine learning models. Therefore, a validation engine has been introduced in the proposed IndiaAnnotate framework which consists of three evaluation criterias: schema consistency, annotation coverage and balance among class distribution, so that it can give out detailed analysis of data quality.

Schema consistency refers to the extent to which each attribute complies to the standard of COCO format annotation specification, whereas annotation coverage refers to the proportion of images which contain valid annotations (both class and bounding box annotation). Another significant statistic of each generated dataset is the distribution of detected object classes.

Since imbalance of class distribution may result in model bias during model training and the class distribution of each detected dataset is presented in **Fig.2**

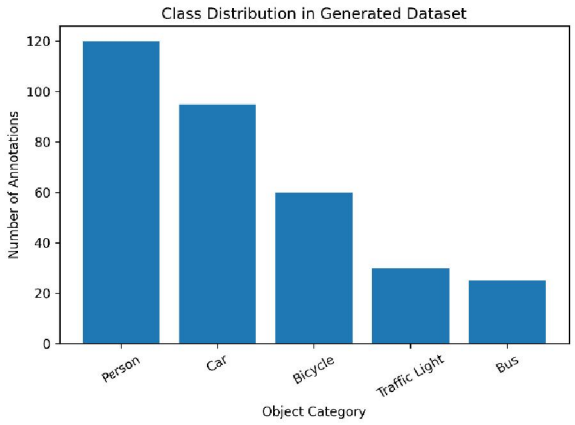

Fig. 2. Class distribution of object categories in the dataset generated by the IndiaAnnotate framework.

By observing the class distribution in Fig. 2 we can detect the possible imbalance of the objects among the different classes. This information will be used in the class imbalance calculation, which will have a role in the quality score calculation mentioned in the methodology. The relatively balanced distribution implies a good representation of object categories.

Comparison with Existing Annotation Approaches

The relative advantage of the proposed system is demonstrated by comparing IndiaAnnotate with traditional annotation schemes used for datasets.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, May 2026
Impact Factor: 8.2

Approach	Annotation Method	Dataset Construction	Dataset Validation
Manual Annotation Tools (LabelMe, CVAT)	Manual labeling	Supported	Not provided
Object Detection Only Pipelines	Automated detection	Partial dataset generation	Not provided
Proposed IndiaAnnotate Framework	Automated object detection	Automated COCO dataset generation	Integrated dataset validation

Table 1. Comparison of the proposed IndiaAnnotate framework with existing annotation approaches.

Human annotation tools are fully reliant on humans; this makes them unsustainable for large data sets. Object detection models help to achieve automated annotation, however, most lack data set construction and validation tools.

The IndiaAnnotate framework proposed herein solves this problem by providing an automated annotation, dataset creation and dataset verification process within a pipeline where a raw image dataset is converted to a validated training-ready dataset in minimum effort.

IV. CONCLUSION

This paper presented the IndiaAnnotate, an automated framework for creation and validation of computer vision datasets from unannotated image collections. It uses an YOLOv8-based object detection module, COCO format dataset creator and a validation module that checks structural and statistical correctness of created datasets.

We conducted experiments and demonstrated that the pipeline is able to convert unannotated images to a structured training dataset in automated way, minimizing required manual workload in comparison to manual annotation procedures. The provided validation module also offers information about the quality of the dataset.

ACKNOWLEDGMENT

I want to express my sincerity to the Vishwakarma Institute of Technology, Pune, to provide an excellent educational environment and resources that make this research possible. I extend my heartfelt thanks to the professors and masters of artificial intelligence and computer science for his guidance and encouragement in this study. His insight into computer -driven analysis and durable technology has greatly influenced my approach to this research.

I will also accept my colleagues and colleagues for their valuable discussion and support, who have enriched my understanding of this topic. Finally, I appreciate authors and researchers whose work has contributed to the basis for this study.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779-788.

[2] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *European Conference on Computer Vision (ECCV)*, Zurich, Switzerland, 2014, pp. 740-755.

[3] Ultralytics, "YOLOv8 Documentation," 2024. [Online]. Available: https://docs.ultralytics.com. [Accessed: Dec. 2024].

[4] A. Ng, "AI Doesn't Have to Be Too Complicated or Expensive for Your Business," *Harvard Business Review*, 2021.

[5] P. Grinberg, "Flask: The Python Micro Framework," 2018. [Online]. Available: https://flask.palletsprojects.com.

[6] N. Sambasivan et al., "Everyone wants to do the model work, not the data work: Data Cascades in High-Stakes AI," in *CHI Conference on Human Factors in Computing Systems*, 2021.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, May 2026
Impact Factor: 8.2

[7] B. Russell, A. Torralba, K. Murphy, and W. Freeman, "LabelMe: A Database and Web-Based Tool for Image Annotation," *International Journal of Computer Vision*, vol. 77, pp. 157-173, 2008.
[8] "JSON Schema Validation: A Vocabulary for Structural Validation of JSON," IETF, 2024. [Online]. Available: https://json-schema.org