e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

AI-ASSISTED IMAGE ANNOTATION SYSTEM: ENHANCING DATA LABELLING FOR COMPUTER VISION

Chetan Parihar*1, Vishal Ghorwade*2

*1,2Pratibha College Of Commerce & Computer Studies Chinchwad, Pune, India.

DOI: https://www.doi.org/10.56726/IRJMETS73213

ABSTRACT

The escalating demand for high-quality labelled data in computer vision often outpaces the capabilities of traditional annotation methods. Manual image annotation is notoriously time-consuming, expensive, and prone to inconsistencies, while fully automated systems frequently falter on domain- specific datasets, requiring extensive corrections. This paper introduces an AI-Assisted Image Annotation System designed to bridge this gap by synergizing human expertise with machine learning efficiency. Our approach integrates the YOLOv8 object detection model to provide initial annotations, which users can rapidly verify and refine. Crucially, the system employs an iterative learning mechanism, progressively enhancing model accuracy based on user corrections. This semi-automated workflow significantly reduces the manual effort and time required for labelling large datasets—by up to 70-80% in our experiments—while maintaining high-quality annotations essential for developing robust computer vision models.

I. INTRODUCTION

The proliferation of real-world artificial intelligence (AI) applications, spanning autonomous driving, medical diagnostics, and industrial automation, critically depends on large, accurately annotated datasets. However, creating these datasets remains a primary bottleneck in AI development. Traditional manual annotation, though considered the gold standard for quality, is inherently slow, costly, and impractical for the scale required by modern deep learning.

Conversely, fully automated annotation systems using pre- trained models often lack the necessary accuracy, particularly when applied to specialized domains, leading to significant manual rework. While active learning strategies aim to optimize labelling efficiency by selecting uncertain samples, they can introduce system complexity and may require continuous retraining efforts unsuitable for all workflows.

To address these limitations, we propose an AI-Assisted Image Annotation System that offers a practical balance between automation and human oversight. Our system leverages the state-of-the-art YOLOv8 object detection model to automate the initial bounding box generation. Users then interact with the system by correcting or validating these AI-generated predictions, providing crucial feedback. This human-in-the-loop process is coupled with an iterative training cycle, allowing the underlying model to adapt and improve its performance specifically for the target dataset with minimal user input required for initial training. This semi-automated, iterative refinement approach significantly accelerates the annotation process compared to manual methods, improves label consistency, and scales effectively for large projects. This paper details the system's architecture, workflow, key features, and evaluates its performance, demonstrating its potential to streamline dataset creation across diverse computer vision domains.

Objectives

The primary goal of this research is to develop and evaluate an AI-Assisted Image Annotation System that significantly enhances the efficiency, accuracy, and scalability of the image labelling process compared to traditional methods.

1) Develop an Efficient and User-Friendly Annotation Tool:

- Implement an intuitive graphical user interface (GUI) that simplifies image navigation, label management, and model interaction.
- Integrate AI-based object detection (YOLOv8) to provide real-time annotation suggestions, drastically reducing manual drawing effort.

2) Enhance Annotation Accuracy via Iterative Learning:

- Employ an iterative training loop where the AI model learns and improves from user corrections over successive cycles.
- Reduce inconsistencies and errors common in purely manual annotation by leveraging AI suggestions and facilitating easy refinement.

3) Ensure Scalability for Large Datasets:

- Design the system to handle large volumes of images efficiently, minimizing the increase in human effort as dataset size grows.
- Enable users to achieve high model accuracy with a limited number of retraining cycles, facilitating rapid annotation of thousands of images.

4) Provide Flexible Data Handling and Compatibility:

- Allow export of labelled datasets in multiple standard formats (e.g., YOLO, COCO, Pascal VOC) to ensure compatibility with common machine learning frameworks.
- Support diverse object classes and annotation tasks suitable for various AI domains.

5) Evaluate System Performance Against Baselines:

- Quantitatively assess the proposed system's speed and efficiency gains compared to traditional manual annotation workflows using relevant metrics.
- Evaluate the accuracy progression of the iteratively trained model on representative datasets.

Research Problem

The field of computer vision relies heavily on high-quality labelled datasets to train deep learning models effectively. However, the process of manually annotating images remains one of the most significant bottlenecks in AI development. Traditional annotation methods require extensive human effort, making them inefficient, time-consuming, and costly. Moreover, inconsistencies in labelling due to human error can negatively impact model performance.

Existing annotation tools either rely entirely on manual efforts or attempt to automate the process using pre-trained models. Fully manual annotation is impractical for large datasets, while automated methods often struggle with domain-specific data, leading to inaccurate predictions and requiring further manual correction. While active learning approaches offer an alternative by selecting uncertain samples for human labelling, they introduce additional system complexity and require continuous retraining, making them difficult to integrate across various models and datasets.

Therefore, there is a critical need for a **hybrid annotation system** that integrates AI-based assistance with human intervention in a way that **minimizes labelling effort while maximizing accuracy and scalability**. The proposed AI- Assisted Image Annotation System addresses this challenge by leveraging a **semi-automated, iterative learning approach** using the YOLOv8 object detection model. By enabling users to label a small number of images, train a model, and iteratively refine predictions, the system significantly reduces annotation time without adding unnecessary complexity.

This research seeks to answer the following key questions:

1. How can AI be leveraged to reduce the time required for image annotation while maintaining high accuracy?
2. What is the optimal balance between AI-assisted labelling and human correction to ensure efficiency and reliability?
3. How can an iterative learning approach improve model predictions over time while minimizing the number of training cycles?
4. How does this system compare with traditional manual annotation and active learning-based methods in terms of speed, accuracy, and usability?

II. METHODOLOGY OVERVIEW

The proposed AI-Assisted Image Annotation System aims to enhance the efficiency of image labelling by combining manual annotation with AI-based object detection. This hybrid approach reduces annotation time while improving accuracy through iterative training cycles. The system leverages YOLOv8, a state-of-the-art object detection model, to assist users in labelling images, refining predictions, and progressively improving the model's performance.

System Architecture

The system consists of two main components:

1. **Frontend (User Interface)**

- A graphical user interface (GUI) for image navigation, labelling, and model control as shown in fig1.
- Tools for adding, renaming, and deleting labels for different object classes.
- A model control section for enabling/disabling AI predictions and adjusting confidence thresholds.
- An export functionality to save labelled data in various formats (e.g., YOLO, COCO, Pascal VOC).

2. **Backend (AI Model & Data Processing)**

- Utilizes YOLOv8 for object detection and iterative learning.
- Includes a training pipeline that allows users to train models on their labelled dataset.
- Implements an AI-assisted annotation process where the model suggests labels, and users refine incorrect predictions.
- Periodically retrains the model to improve accuracy using newly labelled images.

Annotation Workflow

The system follows a structured workflow for efficient image annotation:

Step 1: Select Dataset Folder

- The user selects a folder containing the images that need to be labelled.
- The system loads images into the annotation tool for easy navigation.

Step 2: Manual Labelling of Initial Images

- The user manually annotates a small batch of images (e.g., 10–50 images) to create an initial labelled dataset.
- Bounding boxes are drawn around objects, and labels are assigned.
- The labelled data is saved and used for training.

Step 3: Train AI Model on Initial Labels

- The user initiates model training using YOLOv8 on the labelled dataset.
- The system trains a custom model with a small batch size, ensuring fast initial training.
- Once training is complete, the trained model is saved for inference.

Step 4: Enable Model for AI-Assisted Annotation

- The user enables AI predictions to assist with labelling.
- The model predicts object locations and assigns labels for new images.
- Predictions are displayed to the user for verification.

Step 5: Refinement of Model Predictions

- If the model prediction is incorrect, the user can manually correct it.
- Corrected labels are added to the dataset to improve model accuracy.
- The system allows users to edit, delete, or modify labels as needed.

Step 6: Iterative Model Training

- After annotating more images, the user retrains the model with the expanded dataset.
- With each retraining cycle, the model's accuracy improves, reducing the need for manual corrections.
- Typically, after three training cycles, the model reaches high accuracy, making large-scale annotation much faster.

Step 7: Exporting Labelled Data

- The labelled dataset can be exported in different formats (YOLO, COCO, Pascal VOC) for use in various

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

machine learning frameworks.

- The dataset can be used for further research, model comparison, or deployment.

Key Features of the System

- Real-Time AI-Assisted Annotation: AI provides initial labels, reducing manual effort.
- User-Friendly Interface: Simple UI for navigating images, managing labels, and training models.
- Confidence Threshold Adjustment: Users can set a threshold to control model predictions.
- Iterative Learning: The model improves over time with additional user-corrected annotations.
- Multi-Format Export: Labelled datasets can be exported in formats compatible with major deep- learning frameworks.
- Scalability: Supports large datasets and multiple object classes.

Advantages Over Traditional Annotation Methods

Feature	Traditional Manual Annotation	AI-Assisted Annotation System
Time Efficiency	Slow, labour- intensive	Faster annotation process
Accuracy	High but requires manual effort	High, improves with iterative learning
Automation	None, fully manual	Partially automated with AI
Scalability	Difficult for large datasets	Scales efficiently with retraining
Model Flexibility	No built-in training capability	Trains custom models for better accuracy
Export Formats	Limited	Supports multiple formats (YOLO, COCO, Pascal VOC)

Experimental Setup & Evaluation

To evaluate the effectiveness of the system, experiments were conducted on two datasets:

Dataset 1: Flappy Bird Dataset

- Total dataset size: 174 images
- Number of Classes: 3
- Annotation Time: 26 seconds per image (manual) → Reduced by 70-80% with AI assistance
- Model: YOLOv8-X, batch size = 20, trained on 10 images
- Accuracy: 80% at 50% training

Fig 1: Model trained on a subset of the Flappy Bird dataset, consisting of 10 images. The prediction confidence threshold was set at 0.5

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

Dataset 2: Indian Currency Dataset

- Total Classes: 4
- Total dataset size: 74 images
- Annotation Time: 7 seconds per object (manual) → Reduced by 70-80% with AI assistance
- Model: YOLOv8-m, batch size = 30, trained on 18 images
- Accuracy: 70%

Fig 2: Prediction YOLOv8-n trained on 18 images, 30 batch size, confidence threshold of 0.3

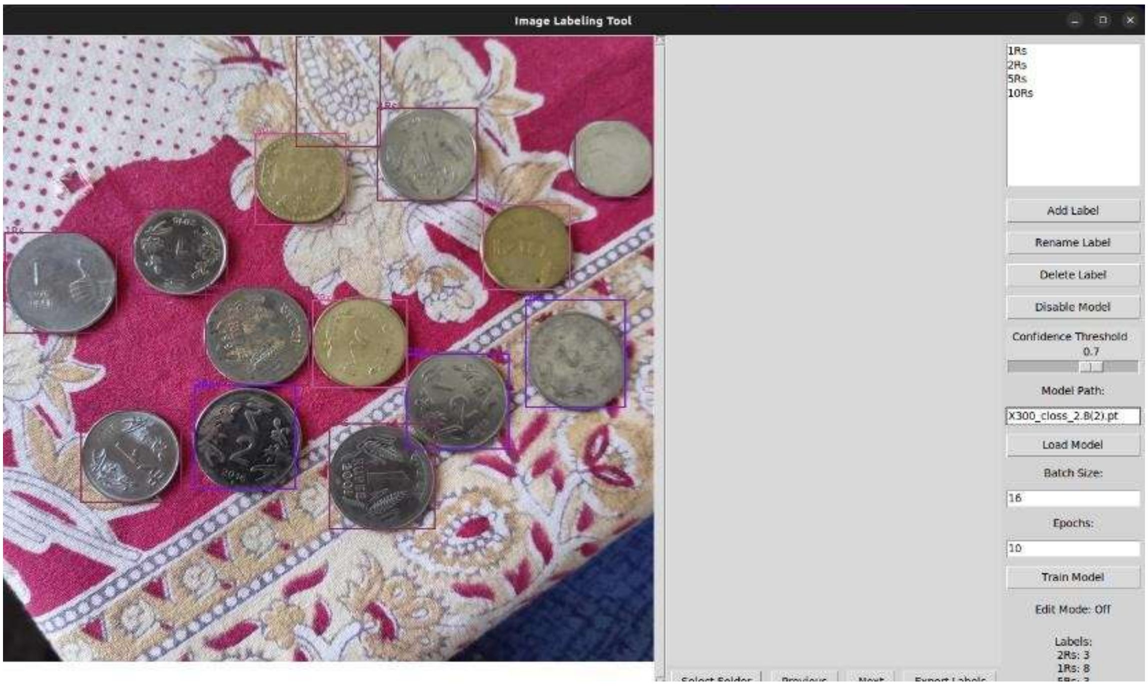

Fig 3: Prediction YOLOv8-m trained on 18 images, 30 batch size, confidence threshold of 0.7

III. CHALLENGES

Despite the advantages of the proposed AI-Assisted Image Annotation System, several challenges must be addressed to ensure its efficiency, scalability, and usability.

1. Initial Model Accuracy and Cold Start Problem

- When the system is first used, the AI model has no prior knowledge of the dataset, leading to poor initial predictions.
- Users must manually annotate a small dataset (10– 50 images) before AI assistance becomes effective.
- The cold start problem can make the system feel ineffective initially, requiring patience and iterative training.

2. Model Confidence and False Positives

- The AI model may detect objects with low confidence or generate false positives, leading to unnecessary corrections by users.
- Setting an appropriate confidence threshold is critical to balance between accuracy and usability.
- Users must continuously adjust thresholds to avoid excessive false detections or missing important objects.

3. Computational Resource Constraints

- Training deep learning models like YOLOv8 requires significant computational power (GPU/TPU), which may not be accessible to all users.
- Running AI-assisted annotation on low-end hardware can lead to slow inference speeds.
- Deploying the system on edge devices (drones, embedded systems, industrial cameras) requires optimizing model size and efficiency.

IV. FUTURE ENHANCEMENTS

As AI technology and annotation tools continue to evolve, several enhancements can be integrated into the **AI-Assisted Image Annotation System** to further improve efficiency, scalability, and usability. Below are some key future enhancements:

1. Integration of Segment Anything Model (SAM)

- **SAM (Segment Anything Model)** is a state-of-the- art AI model developed by Meta that allows users to segment objects in images with minimal effort.
- By integrating **SAM**, the annotation system can **automatically segment objects** instead of just detecting bounding boxes.
- This would be especially useful for applications like **medical imaging, autonomous driving, and industrial defect detection**, where precise segmentation is crucial.
- Users could refine **AI-generated segmentation masks** instead of drawing them manually, saving significant time.

Benefit: More accurate and flexible annotations beyond bounding boxes.

2. Collaborative Annotation System

- Currently, the AI model runs locally on a **single PC**, limiting accessibility to individual users.
- Future versions will allow **collaborative annotation**, where multiple users can annotate images in real-time from different devices.
- A **centralized AI model** will run on a **server or cloud**, and multiple team members can contribute to the annotation process simultaneously.
- **Key features of collaboration:**
  - **Role-based access** (Admin, Reviewer, Annotator)
  - **Real-time model updates** to improve accuracy as users annotate more data
  - **Version control** to track annotation changes

Benefit: Faster dataset annotation for large teams working on massive datasets.

3. Cloud-Based Deployment for Scalability

- Instead of running the model locally, a **cloud-based annotation system** can provide:
  - **On-demand GPU access** for AI-assisted labelling .

- **Secure cloud storage** for large datasets.
- **Remote access** for distributed teams.

- The system could be integrated with cloud platforms like **Google Cloud, AWS, or Microsoft Azure** for high-performance AI inference.

Benefit: Allows seamless scaling for enterprise-level dataset annotation projects.

V. CONCLUSION

The AI-Assisted Image Annotation System presents a significant advancement in dataset creation for computer vision applications. By integrating YOLOv8 for AI-assisted labelling, the system effectively reduces annotation time while maintaining high accuracy through iterative learning and human refinement. This hybrid approach ensures that datasets are labelled efficiently without relying solely on either manual annotation or fully automated systems, both of which have inherent limitations.

One of the key advantages of this system is its ability to scale efficiently. By leveraging AI predictions and allowing manual corrections, the system reduces human effort while continuously improving the model's performance. This approach leads to a 70-80% reduction in annotation time, making large-scale dataset creation more practical and cost- effective. Additionally, the system's ability to export labelled data in multiple formats ensures compatibility with various machine learning frameworks, allowing researchers to experiment with different AI models and determine the best- performing architecture for their specific use case.

Unlike active learning, which introduces complexity by requiring custom strategies for each model, the proposed approach keeps the workflow simple yet effective. With just three training cycles, the model achieves high accuracy, making thousands of images annotatable in significantly less time. The methodology is flexible, adaptable, and scalable, making it suitable for various applications, including medical imaging, surveillance, industrial automation, and autonomous vehicles.

VI. REFERENCES

[1] Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLOv8: Ultralytics Implementation of YOLO for Object Detection and Segmentation. Ultralytics Documentation.

[2] Kirillov, A., Mintun, E., Ravi, N., et al. (2023). Segment Anything Model (SAM). Meta AI Research, arXiv preprint arXiv:2304.02643.

[3] Lin, T. Y., Maire, M., Belongie, S., et al. (2014). Microsoft COCO: Common Objects in Context. European Conference on Computer Vision (ECCV).

[4] Wang, Q., Zhang, Q., Li, L., & Liu, B. (2020). Interactive image segmentation with dilated convolutions. Medical image analysis, 59, 101579

[5] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804. 02767. (For background on YOLO evolution)