Utilization of cloud infrastructure for dataset markup*

Roman Syzonenko[1,†], Svitlana Klymenko[1,*,†] and Volodymyr Hnatushenko[2,†]

[1] *Oles Honchar Dnipro National University, Nauky Ave 72, Dnipro, 49045, Ukraine*
[2] *Dnipro University of Technology, Dmytra Yavornytskoho Ave 19, Dnipro, 49005, Ukraine*

Abstract
The article examines the challenges of creating high-quality, annotated datasets for machine learning, particularly in the field of computer vision. It uses a comparative analysis to develop recommendations for choosing a tool that fits the specific needs of each project. A comprehensive review of existing literature supports the study's conclusions. The paper suggests implementing Label Studio on AWS, utilizing Docker and AWS services, such as S3, RDS and Elastic Beanstalk, to improve scalability, privacy, and cost-efficiency. It offers detailed setup instructions and stresses that selecting an annotation tool should be based on the project's unique requirements, privacy concerns, and collaboration capabilities. The content is enriched with numerous practical examples, making it a valuable resource for researchers and practitioners in the AI field.

Keywords
data labeling, computer vision, cloud platforms, Label Studio, Roboflow, AWS, machine learning

1. Introduction

In the contemporary context of accelerated development in artificial intelligence and computer vision technologies, the imperative for automated visual data analysis is gaining significance. For this type of analysis, it is advisable to use computer vision models such as YOLO or Mask R-CNN [1, 2]. However, it should be noted that particular difficulties and limitations accompany the use of these models.

Firstly, the vulnerability of computer vision models to the quality of their training is evident. This quality is influenced by factors such as the number of training epochs and the quality of the dataset. The number of training epochs is currently more of a formal issue than a practical problem, as server time is becoming cheaper, and some cloud platforms even provide it for free for training various neural networks, albeit with some limitations. The crux of the issue lies in the quality of the dataset, which can be delineated by three factors: the size of the dataset, the diversity of the data in it, and the quality of its labeling. The challenges posed by dataset size and image quality can be addressed through two primary approaches: the acquisition of additional real data or the artificial synthesis of data that meets the requisite criteria. Real data, of course, is preferable as it reflects the actual situation for which the computer vision model is being created [3]. However, obtaining real data, especially if it is peculiar, can be very difficult and therefore expensive. Conversely, one may employ synthesized data, or, as it is alternatively designated, synthetic data. Several methods can be utilized to obtain the desired image. One such method involves the use of copy and paste, in which the necessary information is extracted from the original image and subsequently pasted onto a different background.

Furthermore, synthetic data can be obtained using the methods outlined in article [4], such as random scaling and shifting of the height or width of images, as well as random changes in

* *The 5th International Conference on Information Technologies: Theoretical and Applied Problems (ITTAP-2025) October 22-24, 2025 Ternopil, Ukraine*
[1*] Corresponding author.
[†] These authors contributed equally.
syzonenko_r@365.dnu.edu.ua (R. Syzonenko); klymenko_s@365.dnu.edu.ua (S. Klymenko); vvgnat@ukr.net (V. Hnatushenko)
0009-0007-1023-3894 (R. Syzonenko); 0000-0003-2005-9993 (S. Klymenko); 0000-0003-3140-3788 (V. Hnatushenko)

brightness. In the context of images typical of UAVs, the addition of weather effects such as snow, rain, or fog using masks or generative AI is a viable option. An alternative approach involves the utilization of generative artificial intelligence or the direct application of scenes derived from a graphics or game engine, such as Blender or Unity. Anyway, the training of machine learning models, particularly for object detection, classification, and segmentation tasks, necessitates a substantial quantity of high-quality annotated data. The annotation process, defined as the act of marking images or videos with objects, their respective classes, and coordinates, constitutes a pivotal stage in the development of effective computer vision systems. The process of annotating data to create a dataset, which is essential for analyzing images obtained from a UAV, entails its own unique challenges. These include high image resolution, non-standard angles, changing lighting conditions, and the influx of a substantial volume of information in real time. Synthetic data, however, is not a panacea. Its use requires careful analysis of the generated data. An excessive amount of synthetic data leads to a phenomenon known as the simulation-to-reality (Sim2Real) gap. In this gap, a model trained on synthetic data performs poorly in the real world.

In the context of the preceding discourse on dataset quality, this study concentrates on the second point, that is, the quality of the dataset's labeling. The issue of dataset size, which was previously discussed and is the subject of separate studies, is not addressed in this particular investigation. That is, the creation of specific annotations or markers that are used in training artificial intelligence (AI) and designate regions of interest. These are areas of the image where the object of interest is present and must be identified, classified, or segmented by a computer vision model in the future. In the process of developing machine learning models for computer vision tasks, it is imperative to have high-quality annotated datasets. In particular, when working with video and photo data from UAVs, it may be necessary not only to mark the objects themselves, but also to track the movement of these objects or to classify territories.

There are multiple methods for augmenting a dataset; however, each approach possesses its own set of advantages and disadvantages. The most elementary and economical approach entails the utilization of a pre-trained model for image analysis, with the subsequent exportation of the model's extracted information into the format employed during the training process. While this approach is expeditious and economical, it necessitates an already trained model. That is to say, one must first train a model to create a dataset for the subsequent training of the current model. While this assertion may initially appear counterintuitive, its practical applications are evident. To illustrate, a model is trained on a dataset that is not publicly available. This model is then used to create a new dataset, which is subsequently used to train a new model, either the same or a different architecture. Alternatively, a base model trained on a relatively general dataset can be used to create a more specialized dataset, thereby reducing the size of the new model or enhancing its accuracy. This may be necessary, for example, when intending to use a new model on IoT platforms with limited computing resources. Nevertheless, the efficacy of the outcome is contingent upon the caliber of the pre-trained model. Moreover, it should be noted that the applicability of this method is not ubiquitous.

A more traditional and reliable method for obtaining annotated data is through manual data processing. In this scenario, creating a set of images for annotation is insufficient. Implementing specific annotation creation tools is also necessary. Given that the optimal dataset size for computer vision applications typically ranges from half a thousand to several thousand or tens of thousands of images, the process of creating annotations is often time-consuming and costly, necessitating the involvement of multiple individuals. This necessity gives rise to the imperative for a highly efficient, scalable, and automated infrastructure for the processing and annotation of data. One contemporary solution involves the utilization of cloud-based data annotation platforms such as Roboflow Annotate or Amazon SageMaker Ground Truth. Alternatively, a combination of cloud infrastructure, including Amazon Web Services (AWS), with open platforms for data annotation, such as Label Studio, can be employed. This approach enables the orchestration of teamwork, the integration with data sources, and the preparation of datasets in the requisite format.

2. Analysis of recent research and publications

Numerous solutions exist for data annotation, particularly for videos and images. Well-known platforms include tools such as CVAT, VIA (VGG Image Annotator), LabelStudio, and LabelMe. However, in terms of scalability and flexibility, cloud solutions such as Roboflow or AWS SageMaker Ground Truth are among the most effective solutions for the development of such infrastructures. However, it must be acknowledged that these solutions are not without their own set of drawbacks. For instance, AWS SageMaker Ground Truth has been observed to incur a substantial financial burden when annotating voluminous datasets. Moreover, utilizing Roboflow often results in the dataset's public availability, a circumstance that is regarded as highly unfavorable by some dataset proprietors.

As presented in Article [5], SAINE, an annotation and inference mechanism built on open-source tools, including Label Studio, has been developed to facilitate meta-scientific research. This system enables expert economists to assign hierarchical classifications to scientific articles efficiently. Annotations collected using Label Studio have been demonstrated to enhance the precision and clarity of scientific research workflows, as evidenced by user research. Article [6] presents a series of tools designed for the annotation of datasets. These tools are then subjected to a comparative analysis in an effort to identify the most suitable solution for the designated task, which is the annotation of a visual dataset tailored to meet the specific requirements of the field of gastroenterology. The article discusses annotation tools such as LabelImg, labelme, Visual Object Tagging Tool (VoTT), VGG Image Annotator (VIA), and Computer Vision Annotation Tool (CVAT), listing their advantages and disadvantages. Following a comparative analysis, the authors conclude that the available tools do not meet their requirements. Consequently, they present their own specialized tool, FastCAT, and compare it with CVAT using the GIANA [47] datasets as an example. They also compare it with a separate dataset collected with the assistance of the German clinic, University Hospital Würzburg. The FastCAT algorithm, developed by the authors, demonstrates its superiority to CVAT on both datasets. A comparative analysis of tools for annotating datasets, including LabelImg, VGG Annotator, Label Studio, and Roboflow, is provided in [8]. The evaluation encompasses a range of criteria, including functionality, ease of use, platform support, collaboration capabilities, data types, integration, customization, and scalability. Label Studio is distinguished by its versatility in managing text, audio, and images, offering features for collaboration and customization. Conversely, Roboflow, as evidenced by the analysis results, is particularly adept at cloud projects and seamlessly integrates with machine learning pipelines. A substantial and exhaustive analysis of various annotation tools is provided in [9], wherein the authors not only furnish a comparative table of 25 tools, the data for which these tools are used, the available formats for exporting labeled data, and the techniques that can be used for labeling, but also meticulously categorize these tools by subject areas in which they are most frequently utilized or demonstrate optimal performance.

A further aspect of dataset annotation that merits consideration is the feasibility of collaborative annotation of a dataset by multiple specialists concurrently. This aspect is thoroughly delineated in publication [10], which provides a comprehensive analysis of the problem of parallel dataset labeling. The authors offer their own development — a pipeline that uses Label Maker for labeling and Data Clinic and MLCoach for training machine learning models. A secondary point highlights the use of multiple web-based graphical user interfaces (GUIs) to foster teamwork, noting that a range of specialists are involved in labeling the dataset, especially for large datasets, and in training the model. It is asserted that a straightforward and comprehensible graphical interface is imperative to streamline and enhance their efforts.

3. Problem statement

A comprehensive review of the extant literature reveals a consistent emphasis on the significance of practical labeling tools for machine learning in the field. For instance, comparative analyses of leading tools such as SageMaker Ground Truth and Label Studio emphasize their capacity to support diverse data types and automation, as evidenced in [11]. A review of the extant literature reveals that cloud platforms have been demonstrated to reduce the time and cost of labeling, particularly for voluminous datasets. Moreover, these platforms have been identified as the optimal choice for storing and processing large volumes of images. In this context, Amazon S3 object storage service should be highlighted separately, as it is available as an option for uploading information and storing annotated data in many annotation tools. uses the Libertinus fonts. You may have to install these fonts on your computer. The text below shows how to locally install them.

This paper proposes a workflow for annotating computer vision datasets that meets the following criteria: the dataset remains inaccessible to the public during and after annotation, multiple specialists can work on it simultaneously, it can be tailored to the specific needs of the specialists, and the workflow includes an automated annotation mechanism — preferably utilizing existing tools or a third-party AI model. To clarify the issue further, it is necessary to revisit the literature review and analyze the most popular annotation tools, along with their respective strengths and weaknesses. Examples of such tools include AWS SageMaker Ground Truth, Label Studio, Roboflow, and CVAT. The following section describes each of these components, while Table 1 provides a comprehensive overview of their main features.

Table 1
Key features of selected tools for dataset markup

Tool name	Type	AI-Assistance	End-To-End CV workflow	Deployment options
CVAT	Open-source	Yes (DL-assisted)	No (annotation only)	On-prem / cloud
Label Studio	Open-source	Limited (external ML only)	No	On-prem / self-hosted / SaaS
Roboflow Annotate	Commercial SaaS	Yes (SAM-2, Auto Label)	Yes	Cloud SaaS
Amazon SageMaker Ground Truth	Commercial AWS	Yes (active learning)	Partial	AWS-managed service

Amazon SageMaker Ground Truth is a scalable solution for commercial-grade data labeling. The system offers integration with the broader AWS ecosystem and supports pre-labeling, active learning, and human-in-the-loop workflows. Despite its considerable power, the system has been primarily designed for large organizations and has several drawbacks for academic users. The cost of use can be prohibitive, especially for projects requiring significant annotation efforts. Furthermore, its tight integration with AWS infrastructure imposes limitations on portability and transparency. Furthermore, data must be stored in AWS services such as S3, which has given rise to concerns regarding vendor lock-in and data sovereignty, particularly in jurisdictions with strict data governance policies.

Conversely, Roboflow offers a cloud-based platform that has been optimized for the annotation and training of models. The software's interface is characterized by its user-friendliness, and it incorporates artificial intelligence tools that facilitate the annotation process. However, this convenience is accompanied by a trade-off: a reduction in the level of control over data. As a commercial cloud service, Roboflow typically requires data to be uploaded to external servers, which may not be suitable for projects involving confidential or private data. Additionally, although Roboflow offers a complimentary plan, access to advanced features and the utilization of substantial data volumes typically necessitate a subscription, which may not be suitable for long-term academic endeavors.

In contrast to fully cloud-based solutions, Label Studio is an open-source multimodal annotation tool designed for maximum flexibility and integration. In contrast to the numerous annotation platforms that are constrained to a limited set of data types or annotation modes, Label Studio offers a distinctive feature by supporting images, video frames, text, audio, time series, and combined modalities within a unified framework. Concerning visual data, the software offers annotation features that include bounding boxes, polygons, segmentation masks, and keypoints. The primary benefit of Label Studio is its self-hosting capability, which enables researchers and organizations to utilize the tool either locally or on secure institutional servers. This feature is of particular importance in domains where data privacy, regulatory compliance (e.g., General Data Protection Regulation [GDPR], Health Insurance Portability and Accountability Act [HIPAA]), or intellectual property protection are critical concerns. In contrast to commercial software as a service (SaaS) solutions, which frequently necessitate the upload of sensitive data to externally managed servers, Label Studio enables users to maintain complete ownership and control of their data throughout the annotation process.

For comparison, another fully open-source tool, CVAT, is notable for its extensive features and widespread use, especially for video annotation and object tracking. The open-source nature of its code is a clear benefit; however, its deployment and maintenance are often more complicated. The process usually requires containerization, such as using Docker. It also involves significant system overhead and manual setup. Although CVAT provides robust capabilities for large-scale annotation tasks, its architecture is less modular and adaptable than Label Studio's, especially for multimodal tasks or tasks that combine text and images. These types of tasks are increasingly crucial in interdisciplinary artificial intelligence research.

4. Overview of proposed solution

To address the central issue posed by this article — the creation of a secure and flexible workflow — the decision was made to continue utilizing Label Studio, as it offers greater flexibility in configuration and adaptability in data storage methods compared to Roboflow. A cost-benefit analysis reveals that Label Studio provides a significant advantage over Amazon SageMaker Ground Truth. The former is distributed under the Apache 2.0 license, which means it is freely available. This feature enables research laboratories, nonprofit organizations, and small businesses to employ advanced annotation workflows without incurring license fees. This is particularly beneficial in academic settings, where financial limitations may impede the adoption of commercial annotation platforms. Despite the absence of an integrated AI labeling feature in Label Studio's default configuration, the software offers a versatile API, enabling seamless integration with customized machine learning models. This feature facilitates the prototyping and evaluation of interactive labeling methodologies, such as active learning and model-cycle annotation, within the research framework. It is recommended that users of Label Studio deploy it on a cloud platform, specifically AWS. This approach not only facilitates the attainment of a satisfactory degree of privacy (a benefit not offered by Roboflow) but also streamlines the workload for multiple data specialists. By storing data on AWS S3, users can rest assured that their data will remain private, as AWS ensures data storage on its infrastructure and strictly controls access to this data. The quality of the user experience is contingent upon the accurate configuration of the infrastructure. AWS itself provides users with a considerable degree of flexibility in components and settings, enabling the implementation of a wide range of architectures. As illustrated in Figure 1, the proposed solution is designed with a specific architectural configuration.

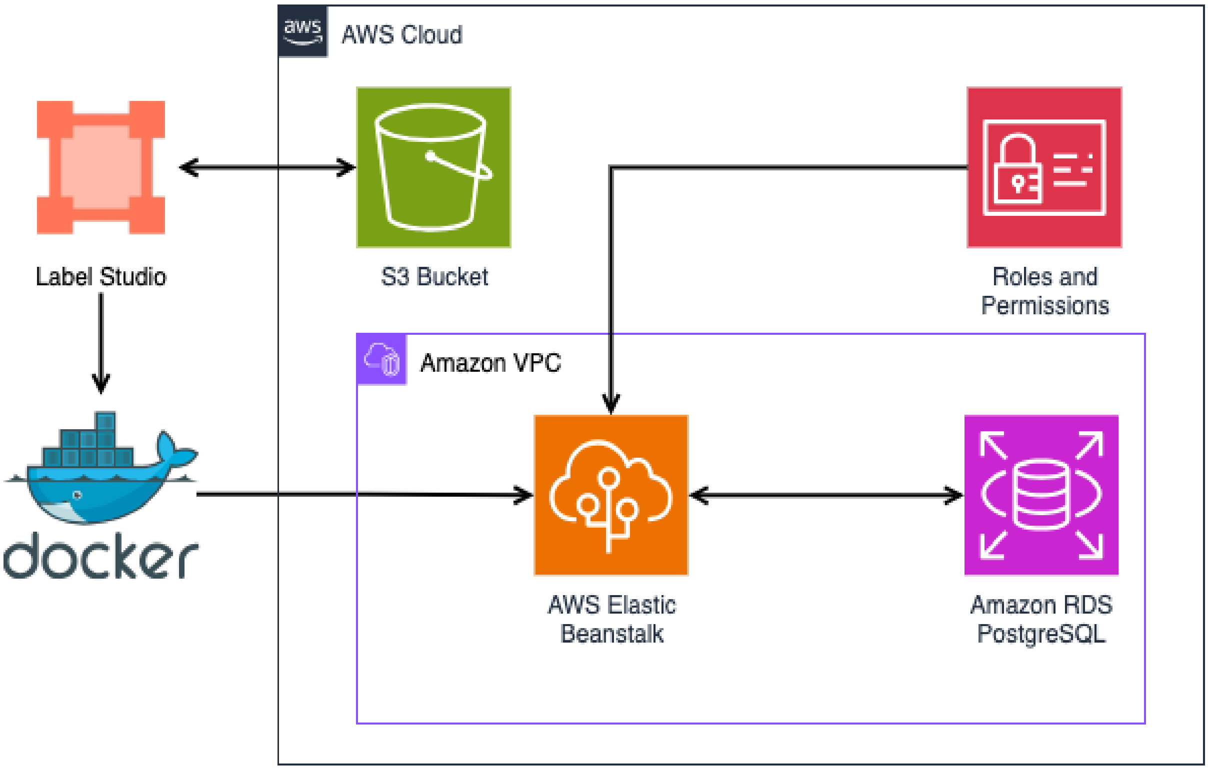

Figure 1: Architecture of the proposed solution.

The following components merit particular attention:

Docker: Given that LabelStudio is open source, one has the option of creating a Docker container for a specific build based on it, or of using the official container. The utilization of containers has been demonstrated to markedly facilitate the delivery process while enabling straightforward and expeditious updates to more recent iterations of the software. To accomplish this objective, it is necessary to either rebuild the container or utilize the official one. It must be acknowledged that certain aspects of using LabelStudio within a container do pose a certain degree of complexity. These aspects will be addressed in greater detail in the forthcoming discussion. The subsequent step entails resolving the issue of Amazon Virtual Private Cloud (VPC), a service designed for managing logically isolated networks. It is possible to use the default VPC, which is available immediately after creating an account. However, for better isolation and more granular control, it would be advisable to create a new VPC specifically for tasks related to Label Studio. The creation of the object in question can be accomplished through the utilization of the graphical initializer and the Resources, thereby facilitating the generation of the object. The VPC configuration is accompanied by a range of additional options, with all other values left at their default settings. This will facilitate the establishment of the requisite network infrastructure for subsequent operations.

Amazon Relational Database Service (RDS) PostgreSQL is a database that will be used to store user data. Within the official LabelStudio container, in addition to the marking tool itself, there is also a database in which user information, such as login and password, is stored. To ensure the integrity of user data and prevent the loss of progress during updates to the container version, it is recommended that this database be isolated from the container itself. To accomplish this objective, it is necessary to establish a distinct Amazon Relational Database Service (RDS) instance with a PostgreSQL database. This is because PostgreSQL is used within the container to manage user data. To utilize the official Docker container with an external database, it is necessary to specify the system variables enumerated in the table. It is imperative to replace the values in the <> with the values obtained from Amazon RDS. In the context of Amazon Relational Database Service (RDS), the values of these variables can be ascertained in the information window after or concurrent with database creation. A separate nuance of creating Amazon Relational Database Service (RDS)

PostgreSQL is the need to select a Virtual Private Cloud (VPC) and subnet. In the context of a standard VPC, the requisite settings must remain unaltered. However, if an alternative VPC is to be employed, it is essential to explicitly specify the desired VPC, in addition to the subnet within which the database will be situated.

Table 2
Environment variables and their values that must be used to utilize the Label Studio Docker container with an external PostgreSQL database

Name	Value
POSTGRE_USER	<postgre_user>
POSTGRE_PASSWORD	<postgre_password>
POSTGRE_HOST	<postgre_host>
POSTGRE_PORT	5432
POSTGRE_NAME	<postgre_db_name>
DJANGO_DB	default

Amazon S3 is an object storage service operated by Amazon. While it is displayed independently from the primary stack, it can be incorporated within the stack through the utilization of Terraform or AWS CloudFormation. To use S3 as a file storage medium for Label Studio, you must establish a connection from Label Studio after initializing the project. To execute this process, navigate to the project's settings menu after its creation. Within the settings menu, locate the tab designated as "Cloud Storage." Once the "Cloud Storage" tab is selected, proceed by clicking on the "Add Source Storage" button. Subsequently, the settings should be configured in accordance with the values delineated in Table 3. This action will initiate the creation of a bucket in which the initial data to be annotated will be stored. To configure the bucket for the exportation of annotated data, first click on "Add Target Storage." Then, fill in the settings in the same way as for the bucket with the source data. When employing a single bucket for both source and annotated data, it is recommended to create folders within the bucket and assign them distinctive names, such as "src" and "target," to distinguish between the source and annotated data, respectively. Subsequently, these folder names should be utilized as prefix values when establishing Source and Target Storage, respectively. Further information regarding additional settings can be found in the official Label Studio documentation, in the "Import & Export" subsection.

The subsequent service necessary for the proposed architecture to function correctly is IAM. IAM is a service that facilitates the management of roles and access policies. It encourages the establishment of well-balanced and granular access policies, enabling the precise calibration of various services. This refinement entails the restriction of integration capabilities to those that are deemed indispensable, thereby ensuring high process isolation. This isolation is paramount for the development of a secure architecture. Turning to a more pragmatic topic, it is imperative to establish a role that will enable Amazon Elastic Beanstalk to access AWS S3. To execute this process, an additional role for the EC2 service must be established and the AWS-managed policy "AWSElasticBeanstalkReadOnly" incorporated into it. This will facilitate access to the EC2 services on which Label Studio will operate, specifically AWS Elastic Beanstalk, which is responsible for deploying the infrastructure. In addition, a policy must be implemented to facilitate the utilization of S3, particularly concerning the designation of specific buckets (or buckets) for the storage of both labeled and unlabeled data. The policy must grant access to retrieve, insert, and delete objects from the bucket, as well as to view the bucket itself. This policy enables users to circumvent numerous access settings within the LabelStudio interface itself. Furthermore, it is necessary to create a role for AWS Elastic Beanstalk that will combine the AWS-managed policies "AWSElasticBeanstalkEnhancedHealth" and "AWSElasticBeanstalkService."

Table 3
Values of parameters used to configure S3 as object storage for LabelStudio

Name	Value
Storage Type	S3
Bucket Name	<name_of_the_S3_bucket>
Bucket Prefix	<bucket_prefix>
Region Name	<name_of_current_region>

AWS Elastic Beanstalk is a service that facilitates the deployment and scaling of web applications and services. This is the primary service that is utilized in this article. Its most salient advantage is the simplicity and flexibility of its settings, a quality that is of critical importance for both commercial and academic use. This feature enables users to prioritize the achievement of objectives over the processes used to accomplish them. To utilize AWS Elastic Beanstalk effectively, it is necessary to create a file required for the correct initialization of the service. This file is known as Dockerrun.aws.json. Within this directory, it is essential to specify the address of the repository that contains the requisite Docker file, in addition to the port mapping. In the case of Label Studio, a single port must be specified: 8080.

Following the creation of the Dockerrun.aws.json file, the subsequent step involves the creation of the AWS Elastic Beanstalk environment. During the initialization process, it is necessary to specify the data presented in Table 4. In the subsequent step, the user must navigate to the service role settings page and select the roles created for the respective services. The following step involves the selection of the VPC and subnets. In the context of a standard VPC, the process is straightforward, and the provided values can be left unchanged. If a discrete VPC is in place, it is necessary to select it and designate private subnets for the purpose of hosting LabelStudio. It is imperative to acknowledge that for optimal functionality, both Amazon Relational Database Service (RDS) and AWS Elastic Beanstalk must operate within a singular Virtual Private Cloud (VPC). It is imperative to turn off the "Enable database" flag, given that the database has been independently created. This is achieved to prevent any potential damage to user data resulting from updates to the AWS Elastic Beanstalk configuration.

Additionally, it facilitates the complete deletion of the AWS Elastic Beanstalk environment without compromising user data integrity. Subsequently, it is necessary to select EC2 security groups. More precisely, a standard group must be chosen, as this option allows incoming traffic from everywhere. The subsequent tab, designated as "Capacity," requires the selection of the desired environmental configuration. The optimal settings are delineated in Table 5; however, they can be modified based on the user's requirements. In the context of employing a load balancer, it is imperative to designate its subnets as public, a configuration that is necessary to facilitate access to the load balancer from the Internet. In the final step, maintaining the default values, navigate to the main page and select the "Add environment property" option located within the "Platform software" submenu. Subsequently, a field will appear in which system variables must be entered. These variables are derived from the data presented in Table 2.

Table 4
Values of parameters used to configure AWS Elastic Beanstalk for hosting LabelStudio

Name	Value
Environment tier	web server environment
Platform	Docker
Platform branch	Docker running on 64bit Amazon Linux 2023
Platform version	4.7.0 (Recommended)
Application code	Upload your code
Source code origin	Local file
Choose file	Dockerrun.aws.json
Region Name	<name_of_current_region>

Table 5
Values of parameters used to configure AWS Elastic Beanstalk, tab Capacity, for hosting LabelStudio

Name	Value
Environment type	Load balanced
Min instances	1
Max instances	2
Fleet composition	On-Demand instances
Instance types	t3.medium

Following the confirmation of the environment's creation, the process of its immediate initialization will commence. This process can be monitored using AWS tools that are integrated into the system. The result will be an environment accessible from the Internet, which will allow several specialists to use Label Studio simultaneously to label the dataset.

5. Conclusions

In summary, the study systematically addresses the urgent issues of dataset annotation for computer vision by reviewing key annotation tools and proposing a scalable, cloud-based architecture. A thorough analysis of contemporary solutions such as AWS SageMaker Ground Truth, Roboflow, CVAT, and notably Label Studio underscores the importance of flexibility, privacy, and cost-effectiveness in the selection of tools and workflows. The literature review stresses the importance of selecting the appropriate dataset labeling tool, and the problem section discusses the advantages and disadvantages of standard tools, offering recommendations for choosing the best option based on specific project needs. The proposed setup, which utilizes Docker on AWS with services such as Amazon S3 for secure data storage and Amazon RDS for user management, introduces an innovative approach that emphasizes data sovereignty, team productivity, and adaptability to project requirements. The architecture is built on the modularity and self-hosting abilities of open-source software, complemented by the scalability and security features of cloud services. This combined framework allows research teams and organizations to create customized annotation environments tailored to their specific needs and goals. Practical recommendations demonstrate that this integration not only protects sensitive data and ensures regulatory compliance but also improves workflow automation and multi-user collaboration. This solution offers a pragmatic, adaptable, and dependable approach to dataset annotation, suited to current and future needs in AI research. It serves as a valuable asset for researchers and practitioners alike.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly in order to: Grammar and spelling check. After using these tool, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] V. Hnatushenko, V. Kashtan, O. Kazymyrenko. Information technology for detecting cars on aerial imaging using a modified YOLO-OBB architecture. MoDaST 2025: Modern Data Science Technologies Doctoral Consortium, June, 15, 2025, Lviv, Ukraine, Pp. 293-304. URL: https://ceur-ws.org/Vol-4005/paper20.pdf.

[2] V. Kashtan, V. Hnatushenko, D. Babets, K. Cyran, K. Wereszczyński. Hybrid quantum CNN-based information technology for building semantic segmentation in aerial imagery. PhD Workshop on Artificial Intelligence in Computer Science at 9th International Conference on Computational Linguistics and Intelligent Systems (CoLInS-2025), May 15–16, 2025, Kharkiv, Ukraine. Pp.150-162. URL: https://ceur-ws.org/Vol-4015/paper11.pdf.

[3] V. Vysotska, N. Sharonova, K. Smelyakov, E. Vakulik, O. Filipov, R. Kotelnykov, Fast color images clustering for real-time computer vision and AI system, CEUR Workshop Proceedings, Vol-3664, 2024, pp. 161–177.

[4] A. Bokhonko, N. Melnykova, Y. Patereha, Comparative analysis of data augmentation methods for image modality, Scientific Journal of the Ternopil National Technical University, 2024, 1(113), pp.16-26. doi: 10.33108/visnyk_tntu2024.01.046.

[5] M. Aqeel, P. Norouzzadeh, A. Maazallahi, S. Tutun, G.R. Miab, L.A. Dehailan, D. Stoeckel, E. Snir, B. Rahmani, Dental cavity analysis, prediction, localization, and quantification using computer vision, AccScience Publishing, Vol. 1, Issue 3, 2024, pp. 80-88. doi: 10.36922/aih.3184.

[6] A. Krenzer, K. Makowski, A. Hekalo, D. Fitting, J. Troya, W. Zoller, A. Hann, F. Puppe, Fast machine learning annotation in the medical domain: a semi-automated video annotation tool for gastroenterologists, BioMed Eng OnLine 21, 2022. doi: 10.1186/s12938-022-01001-x.

[7] Y. Guo, B. Matuszewski, Giana polyp segmentation with fully convolutional dilation neural networks, Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, Volume 4: GIANA; Prague, Czech Republic, 2019, pp. 632–641. doi: 10.5220/0007698806320641.

[8] P. D. Kale, M. Tanvi, K. Ankita, Sh. Samrudhi, J. Shweta, Comparative analysis of image annotation tools: Label img, vgg annotator, Label Studio, and Roboflow, Journal of emerging technologies and innovative research (JETIR), Volume 11, Issue 5, 2024, pp. 398 – 403. URL: https://www.jetir.org/view?paper=JETIR2405D59.

[9] Ch. Moschidis, E. Vrochidou, G. A. Papakostas, Annotation tools for computer vision tasks, Proceedings Volume 13517, Seventeenth International Conference on Machine Vision (ICMV 2024), 2024. doi: 10.1117/12.3055065.
[10] T. Chavez, Zh. Zhao, R. Jiang, W. Koepp, D. McReynolds, P. H. Zwart, D. B. Allan, E. H. Gann, N. Schwarz, D. Ushizima, E.S. Barnard, A. Mehta, S. Sankaranarayananj, A. Hexemera, A machine-learning-driven data labeling pipeline for scientific analysis in MLExchange, Journal of applied crystallography, Volume 58, Part 3, 2025, pp. 731 – 745. doi: 10.1107/S1600576725002328.
[11] M. Mots'oehli, Assistive Image Annotation Systems with Deep Learning and Natural Language Capabilities: A Review, Computer Vision and Pattern Recognition, 2024. doi: 10.48550/arXiv.2407.00252.