

# MLOps Approach for Automatic Segmentation of Biomedical Images

Oleh Berezsky<sup>a</sup>, Oleh Pitsun<sup>a</sup>, Grygoriy Melnyk<sup>a</sup>, Yuriy Batko<sup>a</sup>, Petro Liashchynskyi<sup>a</sup>, Mykola Berezkyi<sup>a</sup>

<sup>a</sup> West Ukrainian National University, 11 Lvivska st., Ternopil, 46001, Ukraine

## Abstract

When using artificial intelligence systems for processing medical images, a large amount of software libraries, data and cloud computing is required. Implementing deep learning elements in CAD is a complex process and applying DevOps can help speed up this process. The implementation of DevOps approaches in the field of machine learning differs from the operations with standard programs; therefore the development of MLOps approaches to the implementation of deep learning elements for the analysis of biomedical images is an actual task. The developed pipeline allows scientists and specialists to use the findings in this article to launch projects based on machine learning and focus on model development rather than the process of setting up the environment. This paper provides examples of improved MLOps pipelines that can be used for solving problems of automatic image segmentation and evaluating the quantitative characteristics of microobjects.

## Keywords 1

Machine learning, MLOps, biomedical images, programming.

## 1. Introduction

Every year, software systems increasingly use machine learning elements. Despite the great demand for neural networks, there is still a need for programmers with specialized knowledge including the knowledge of development and system administrators. Special MLOps approaches are applied to speed up the software development process and increase its reliability and ease of software support. The purpose of this work is to develop MLOps approaches for automatic segmentation of histological and immunohistochemical images and evaluate quantitative characteristics of cell nuclei.

MLOps approaches are developed to efficiently and reliably deploy infrastructure for running machine learning elements and provide convenient and continuous delivery and deployment of program code on cloud systems. This is a relatively new industry that requires the development of solutions for specific subject area. Thus, in this paper, we consider the processing of biomedical images.

Usually, machine learning models are developed at the local level, which does not allow one to quickly run the code on any other computer system for data processing on the basis of machine learning. In most cases, such developments are used at the level of specialized laboratories and do not become widely used. However, modern hardware and cloud computing make it possible to use local developments on an industrial scale. The development of specific pipelines allows automating the process of deploying software code and increasing the efficiency of this process. Applying MLOps approaches for software development can help to get the following advantages:

- less time for preparing and launching machine learning models;
- scalability;
- reduction of the number of errors and elimination of contradictory situations;

---

IDDM'2023: 6th International Conference on Informatics & Data-Driven Medicine, November 17 - 19, 2023, Bratislava, Slovakia  
EMAIL: ob@wunu.edu.ua (A. 1); o.pitsun@wunu.edu.ua (A. 2); mgm@wunu.edu.ua (A. 3); bum@wunu.edu.ua (A. 4); p.liashchynskyi@st.wunu.edu.ua (A. 5); mykolaberezkyi@gmail.com (A. 6);  
ORCID: 0000-0001-9931-4154 (A. 1); 0000-0003-0280-8786 (A. 2); 0000-0003-0646-7448 (A. 3); 0000-0002-6732-4865 (A. 4); 0000-0002-3920-6239 (A. 5); 0000-0001-6507-9117 (A. 6)



© 2023 Copyright for this paper by its authors.  
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).  
CEUR Workshop Proceedings (CEUR-WS.org)

- program automation;
- reduction of possible risks.

Currently, there are already a large number of tools that allow you to deploy infrastructure, such as terraform. Mechanisms for continuous code delivery and deployment are also available. However, most of these mechanisms are used in DevOps tasks.

The scientific novelty of this work lies in the development of MLOps workflow for automatic segmentation of biomedical images using the elements of deep machine learning.

The purpose of our research is to improve the existing mechanisms for machine learning tasks.

The object of our research is the processes of automatic creation of infrastructure for microscopic image processing.

The subject of the research is DevOps practices for the creation of CI/CD pipelines.

## 2. Literature review

In [1], the authors emphasized on the lack of regulatory documents for MLOps and offered their own analysis and classification of the existing documents. Based on the conducted analysis, they proposed a 10-step pipeline.

Sajid Nazir et al. conducted a detailed analysis of artificial intelligence tools for biomedical image processing, using deep machine learning in [2]. In particular, the authors carried out an in-depth analysis of artificial intelligence tools when investigating breast cancer.

In [3], unsolved problems in machine learning relating to the analysis of health preserving means were highlighted. The authors paid considerable attention to the problem of generating datasets for machine learning process.

In work [4], the authors analyzed the problem of organizing interaction between specialists in IT field to solve problems based on machine learning. Therefore, the development of unified pipelines for software deployment is currently one of the most relevant problems in the field of machine learning.

Adrien Bennetot et al. in [5] considered artificial intelligence tools applied on biomedical use case applications. The authors analyzed both standard models of neural networks and the latest ones such as transformers. Due to the analysis of trends in machine learning, modern diagnostic tools are defined. However, there is a need to reduce the complexity of the software configuration process and configure the interaction between different technologies.

In [6-9], approaches for the implementation of DevOps as tools for processing biomedical images were highlighted, which made it possible to create the main elements of the pipeline.

In [10], the authors presented a pipeline for the classification of biomedical images and the structure of a convolutional neural network for the classification of immunohistochemical images. In work [11], an approach for evaluating the quantitative characteristics of microobjects for diagnosis was proposed. The structure of the u-net neural network for automatic segmentation of biomedical images was presented in [12].

The analysis of the above publications has shown that the development of a pipeline, which can be used in deep machine learning tools and algorithms for processing biomedical images, is an urgent task.

## 3. Problem statement

To develop a unified approach for automatic segmentation and evaluation of quantitative characteristics of microobjects, it is necessary to:

- analyze the existing tools for implementing MLOps pipeline;
- select the main components of the pipeline;
- develop a pipeline for processing images with elements of artificial intelligence.

## 4. Analysis of MLOps tools and platforms

Table 1 shows the results of the analysis of MLOps tools in computer vision. The main criteria for evaluating the existing tools are the following ones:

- image classification;
- image segmentation;
- image tagging.

**Table 1**

Results of the analysis of MLOps tools

Software system	Image classification	Image tagging	Image segmentation
Nyckel [13]	+	+	-
Ximilar [14]	+	+	-
Hasty [15]	-	+	+
Levity [16]	+	+	-

Ximilar focuses on systems development with computer vision elements. This software can be considered as an API developer for business. The main emphasis is laid on image recognition and detection of elements in the image. In addition, this program has a convenient means of visually setting up components for the operation of neural networks.

The Nyckel software system processes images and text. Nyckel is characterized by the mechanism for running learning models in a short time and by having a mechanism that makes it possible to use a small amount of input data to get started. The presence of an API allows integration with third-party services, which makes the program more flexible.

The Roboflow platform specializes in computer vision. Roboflow uses cloud technologies to develop workflow management systems for image processing [17]. The Levity software system is implemented using the "no-code" approach, which makes it possible to develop the necessary pipelines and systems for solving problems in many fields, both business and science. In [18], Multi-Domain Object Detection Benchmark was proposed using the Roboflow platform.

In [19], the authors analyzed modern software systems for developing programs with machine learning elements, and highlighted such a stage as "validation" for optimizing the operation of neural network models.

Hasty is used to annotate objects in an image and generate datasets. The main functions of this software product include:

- classification;
- tagging;
- object detection;
- instance segmentation;
- panoptic segmentation;
- attribute prediction.

The main requirements for systems using MLOps approaches were given in works [20-22].

So, modern tools for building a pipeline have a standard set of components, but not all systems have the necessary functionality for image segmentation.

## 5. MLOps workflow for biomedical image segmentation

This section presents a pipeline for automatic segmentation of images using Unet. The key stage in this process is generation and preparation of data (images). This is one of the key differences compared to analogues.

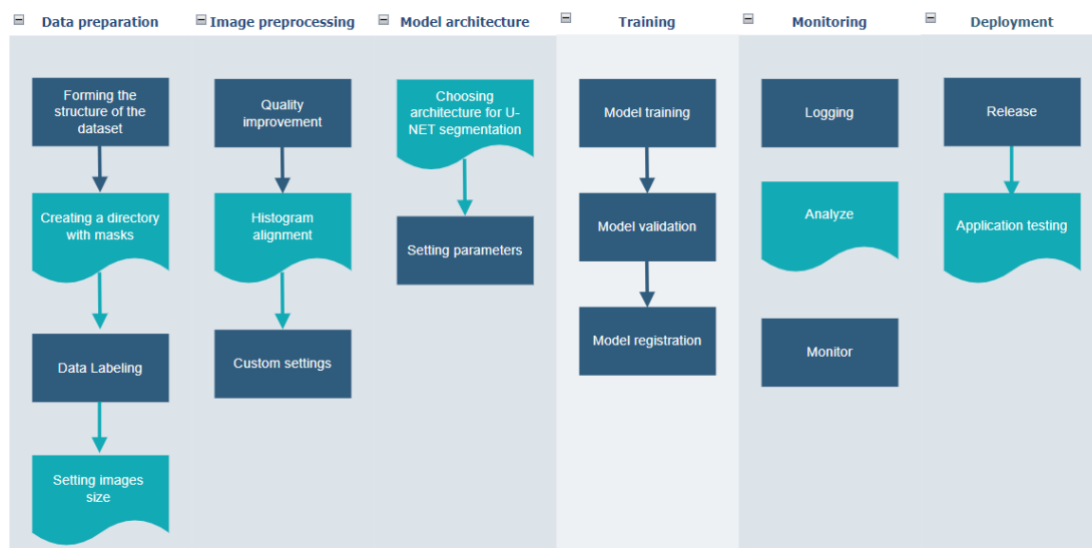
MLOps workflow consists of 3 main components:

1. Build.
2. Deploy.
3. Monitor.

The stage of data preparation includes the following steps:

- creating the directory structure for training and test samples (for example, "original", "masks");
- creating the internal directory structure for storing image masks;
- Data Labelling. The stage includes the rules for creating file names (for example, the suffix "\_mask" is added to the mask);
- changes in file size and other parameters.

MLOps-workflow for biomedical images segmentation is shown in Figure 1.



**Figure 1:** MLOps workflow for biomedical images segmentation.

Image processing is an important stage because we are developing an image processing pipeline. This stage also includes the process of histogram alignment and changes in image parameters depending on the settings.

U-net is used for image segmentation. This is a modern approach that makes it possible to use deep learning. At the same time, it is necessary to create the architecture and select hyperparameters for u-net.

After creating the architecture, it is necessary to conduct training and validate the model.

The monitoring stage is one of the key stages in DevOps approaches and is aimed at analyzing the system performance.

Deployment is necessary for software release and for the use in real conditions.

## 6. CI/CD pipeline for evaluating the quantitative characteristics of microobjects

The module for evaluating the quantitative characteristics of microobjects is an important component of the software system. Unlike other modules, this module can frequently change the code.

This is due to the need to set parameters for different types of images. To automate the process of transferring parameter settings, it is proposed to use a separate `deploy.yml` file (Figure 2).

```
1  name: Build & Deploy
2  on:
3    push:
4      branches: [main]
5
6  jobs:
7    deploy:
8      runs-on: ubuntu-latest
9      steps:
10     - name: Deploy Cell Parameters
11       uses: actions/checkout@v3
12       with:
13         host: ${secrets.SSH_HOST} # IP address of the server you wish to ssh into
14         key: ${secrets.SSH_KEY} # Private or public key of the server
15         username: ${ secrets.SSH_USERNAME } # User of the server you want to ssh into
16         passphrase: ${ secrets.SERVER_PASSPHRASE }
17
18     script: |
19       mkdir experiments
20       cd experiments
21       git clone https://github.com/olehpitsun/AutoImageSegmentation.git
22       echo 'Deployment successful to digitalsal ocean'
23
```

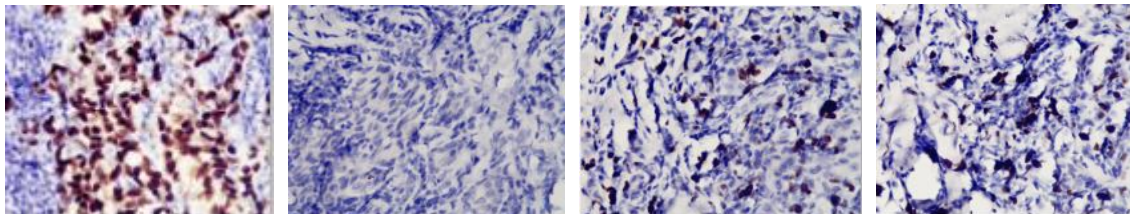
**Figure 2:** Configuration file `deploy.yml`.

In addition to the necessary entries for connection to the cloud server, this example shows the path to the repository with the software code for launching the project of evaluating the quantitative characteristics of microobjects.

## 7. Peculiarities of using the Infrastructure as Code approach

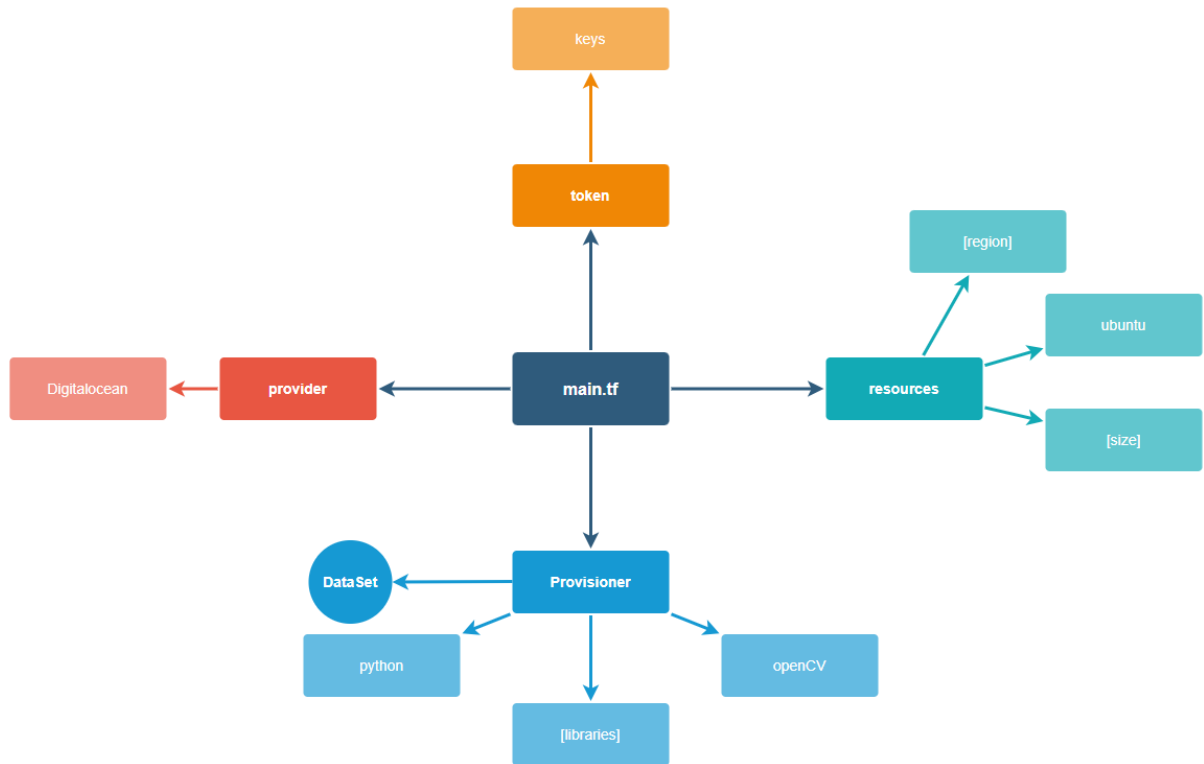
Infrastructure as Code is a modern approach for the development and implementation of software, which makes it possible to write all the necessary elements of the server environment as software code. This is especially convenient for solving problems with elements of deep learning. As a tool, terraform is chosen, which allows one to use a large number of providers to deploy the project on various cloud services, such as AWS, digitalocean, Azure, etc.

An example of biomedical images is shown in Figure 3.



**Figure 3:** Example of biomedical images

Configuration file structure for infrastructure deployment used in a project is shown in Figure 4.



**Figure 4:** Configuration file structure for infrastructure deployment.

Digitalocean is chosen as the provider for conducting experiments. To connect to the cloud storage, a token and ssh-keys are used as standard. Ubuntu server is chosen as the operating system. After deploying the main environment, you need to install the necessary software and download the dataset for further processing.

Software code is updated using the CI/CD mechanism and taking github actions.

The minimum requirements for the developed system are as follows:

RAM – 4GB

Disk Space – 25GB

1000 GB transfer

OS – Ubuntu 18.X

## 8. Conclusions

1. According to the comparative analysis, the advantages and disadvantages of the existing systems with pipelines for automatic segmentation are highlighted. It is found that not all the systems have the necessary functionality.

2. The MLOps workflow is developed for the segmentation of biomedical images based on deep learning with Unet elements.

3. CI/CD pipeline is developed for software code delivery and deployment for evaluating the quantitative characteristics of microobjects.

4. The developed workflow can be a prototype not only for image segmentation programs, but also for solving other problems in another subject area.

## 9. References

- [1] M. Testi et al., "MLOps: A Taxonomy and a Methodology," in *IEEE Access*, vol. 10, pp. 63606-63618, 2022, doi: 10.1109/ACCESS.2022.3181730.
- [2] Nazir, Sajid, Diane M. Dickson, and Muhammad Usman Akram. "Survey of explainable artificial intelligence techniques for biomedical imaging with deep neural networks." *Computers in Biology and Medicine* (2023): 106668. <https://doi.org/10.1016/j.compbimed.2023.106668>
- [3] Dhar, Tribikram, Nilanjan Dey, Surekha Borra, and R. Simon Sherratt. "Challenges of Deep Learning in Medical Image Analysis—Improving Explainability and Trust." *IEEE Transactions on Technology and Society* 4, no. 1 (2023): 68-75. <https://doi.org/10.1109/TTS.2023.3234203>
- [4] Vega, Carlos, Miroslav Kratochvil, Venkata Satagopam, and Reinhard Schneider. "Translational challenges of biomedical machine learning solutions in clinical and laboratory settings." In *International Work-Conference on Bioinformatics and Biomedical Engineering*, pp. 353-358. Cham: Springer International Publishing, 2022. [https://doi.org/10.1007/978-3-031-07802-6\\_30](https://doi.org/10.1007/978-3-031-07802-6_30)
- [5] Bennetot, Adrien, Ivan Donadello, Ayoub El Qadi, Mauro Dragoni, Thomas Frossard, Benedikt Wagner, Anna Saranti et al. "A Practical guide on Explainable AI Techniques applied on Biomedical use case applications." *arXiv preprint arXiv:2111.14260* (2021). <http://dx.doi.org/10.2139/ssrn.4229624>
- [6] Granlund, Tuomas, Vlad Stirbu, and Tommi Mikkonen. "Towards regulatory-compliant MLOps: Oravizio's journey from a machine learning experiment to a deployed certified medical product." *SN computer Science* 2, no. 5 (2021): 342. <https://doi.org/10.1007/s42979-021-00726-1>
- [7] Reddy, Manjunatha, Brahmanand Dattaprakash, Sandesh Kammath, Subramanya Kn, Sumathra Manokaran, and Rangaswamy Be. "Application of mlops in prediction of lifestyle diseases." *ECS Transactions* 107, no. 1 (2022): 1191. DOI 10.1149/10701.1191ecst
- [8] Jain, Archit, Adarsh Malviya, Disha Bajaj, Revati Bhavsar, and Amit Savyanavar. "Brain Tumor Detection using MLOps and Hybrid Multi-Cloud." In *2022 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS)*, pp. 1-6. IEEE, 2022. <https://doi.org/10.1109/ICBDS53701.2022.9936020>
- [9] Stirbu, Vlad, Tuomas Granlund, and Tommi Mikkonen. "Continuous design control for machine learning in certified medical systems." *Software Quality Journal* 31, no. 2 (2023): 307-333. <https://doi.org/10.1007/s11219-022-09601-5>
- [10] Berezsky, Oleh, Oleh Pitsun, Grygory Melnyk, Yuriy Batko, Bohdan Derysh, and Petro Liashchynskiy. "Application Of MLOps Practices For Biomedical Image Classification." In *IDDM*, pp. 69-77. 2022.
- [11] Berezsky Oleh, Pitsun Oleh, Grygoriy Melnyk, Tamara Datsko, Ivan Izonin, and Bohdan Derysh. "An Approach toward Automatic Specifics Diagnosis of Breast Cancer Based on an Immunohistochemical Image." *Journal of Imaging* 9, no. 1 (2023): 12. <https://doi.org/10.3390%2Fjimaging9010012>
- [12] Berezsky, Oleh, Pitsun Oleh, Bohdan Derysh, Ihor Pazdriy, Grygory Melnyk, and Yuriy Batko. "Automatic segmentation of immunohistochemical images based on U-net architecture." In *2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT)*, vol. 1, pp. 29-32. IEEE, 2021. <https://doi.org/10.1109/CSIT52700.2021.9648669>
- [13] Nyckel URL: <https://www.nyckel.com/>
- [14] ximilar URL: <https://www.ximilar.com/>
- [15] hasty URL: <https://hasty.cloudfactory.com/>
- [16] levity URL: <https://levity.ai/>
- [17] Alexandrova, Sonya, Zachary Tatlock, and Maya Cakmak. "RoboFlow: A flow-based visual programming language for mobile manipulation tasks." In *2015 IEEE*

- International Conference on Robotics and Automation (ICRA), pp. 5537-5544. IEEE, 2015. <https://doi.org/10.1109/ICRA.2015.7139973>
- [18] Ciaglia, Floriana, Francesco Saverio Zuppichini, Paul Guerrie, Mark McQuade, and Jacob Solawetz. "Roboflow 100: A Rich, Multi-Domain Object Detection Benchmark." arXiv preprint arXiv:2211.13523 (2022). <https://doi.org/10.48550/arXiv.2211.13523>
- [19] Moreschini, S., Lomio, F., Hästbacka, D., & Taibi, D. (2022, March). MLOps for evolvable AI intensive software systems. In 2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER) (pp. 1293-1294). IEEE. <https://doi.org/10.1109/SANER53432.2022.00155>
- [20] Kreuzberger, Dominik, Niklas Kühl, and Sebastian Hirschl. "Machine learning operations (mlops): Overview, definition, and architecture." IEEE Access (2023). <https://doi.org/10.1109/ACCESS.2023.3262138>
- [21] Kumara, Indika, Rowan Arts, Dario Di Nucci, Willem Jan Van Den Heuvel, and Damian Andrew Tamburri. "Requirements and Reference Architecture for MLOps: Insights from Industry." (2022).
- [22] Recupito, Gilberto, Fabiano Pecorelli, Gemma Catolino, Sergio Moreschini, Dario Di Nucci, Fabio Palomba, and Damian A. Tamburri. "A multivocal literature review of mlops tools and features." In 2022 48th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), pp. 84-91. IEEE, 2022.