Best Practices and Guideline Updated for Developers and Providers of AI-based Image Analytics Services

iMagine Deliverable 4.4

03/03/2025

Abstract

iMagine is a 36-month-long project to serve aquatic researchers with a portfolio of 'free at point of use' image datasets, high-performance image analysis tools empowered with Artificial Intelligence (AI). The iMagine services are built on top of the AI4OS Platform that allows transparent training, sharing, and serving of Machine learning (ML) and Deep Learning (DL) applications. This document is an update to the original D4.1 document created at the beginning of the project, providing good practice usage guides and pointers to documentation relating to the baseline technology of the iMagine AI platform.

Document Description

D4.4 Best Practices and Guideline Updated for Developers and Providers of AI-based Image Analytics Services			
Work Package 4			
Due date	28/02/2025	Actual delivery date:	03/03/2025
Nature of document	Report	Version	1.0
Dissemination level	Public		
Lead Partner	CSIC		
Authors	Ignacio Heredia (CSIC), Valentin Kozlov (KIT)		
Reviewers	Dick Schaap (MARIS), Marco Rorro (EGI)		
Public link	https://zenodo.org/records/14961559		
Project website	https://imagine-ai.eu/		
Keywords	Aquatic sciences, iMagine, AI platform, best practices		

Revision History

Issue	Item	Comments	Author/Reviewer
V 0.1	Draft version	Table of Content	I.Heredia (CSIC), V.Kozlov (KIT)
V 0.2	Revised version	First full draft, revised internally by co-authors	I.Heredia (CSIC), V.Kozlov (KIT), A. Calatrava (UPV), F. Alibabaei (KIT)
V 0.3	Revised version	Reviewed	I.Fava (EGI), G.Moltó (UPV), E.Azmi (KIT), J. O. Irrison (IMEV), M. Laviale, I. Blanquer (UPV), H. Bayındır (TUBITAK)
V 0.4	Revised version	Updated after the review	I.Heredia (CSIC)
V 1.0	Submitted version		Andrea Anzanello (EGI)

Table of content

List of figures

Introduction

Purpose of the document

The iMagine image analysis services build on the common 'iMagine AI Platform' established by Q1 2023, which allows transparent training, sharing, and serving of Machine learning (ML) and Deep Learning (DL) applications. The iMagine AI platform is built on top of the AI4OS stack, a set of services developed and successfully used in various EC initiatives over the past years, particularly the AI4EOSC project[1]. This document aims to offer a consolidated view of the iMagine AI platform by collecting all the documentation and good practice guides available about AI4OS for developing and providing AI-based image analytics services. When needed, we will discuss the customisations in place specifically tailored to the iMagine AI platform.

The iMagine AI framework offers a portfolio of services for AI model development, training, and deployment to be adopted by researchers in aquatic sciences. The collected practices and guidelines provide information on how to develop and operate AI-based image analytics services, which are essential to efficiently producing good outcomes.

Scope of the document

The document contains all the necessary information for iMagine users to use the AI4OS services[2] efficiently. It overviews the services and workflows, providing pointers to the appropriate online documentation for further in-depth details. This will make the document relevant throughout the project, as the online documentation is continuously updated to reflect the latest best practices, guidelines, and tutorials.

Structure of the document

The deliverable is organised as follows: the main components of the platform are listed with corresponding entry points and the online documentation. Then, targeted user roles and workflows are described referring to the platform components. Typical usage scenarios follow this. In the end, the communication channels between users and the platform support are described.

[1] https://ai4eosc.eu/
[2] https://ai4os.eu/

The iMagine AI platform

The iMagine AI platform builds on the services provided by AI4OS for artificial intelligence (AI), machine learning (ML), and deep learning (DL). Therefore, references in this guideline point to available information and documentation about these services and workflows.

How to access

Several services are relevant for users:

- **Project's Homepage** - https://www.imagine-ai.eu
  A high-level overview of the iMagine AI project, which is responsible for the iMagine platform.
- **Documentation** - https://docs.ai4os.eu
  This is the main source of knowledge on how to use the platform. Always refer to it in case of doubt. The documentation covers both user-related documentation and technical notes about the platform itself.
- **YouTube channel** - https://www.youtube.com/@ai4eosc
  YouTube channel of the AI4EOSC project, which is responsible for developing the AI4OS stack, with a number of video tutorials and examples about using the AI4OS stack.
- **EGI CheckIn**- https://aai.egi.eu
  This is the authentication manager of the platform, where iMagine users should register to get access to the authentication-only resources, like Dashboard deployments or the Nextcloud storage (see below).
- **Dashboard** - https://dashboard.cloud.imagine-ai.eu
  This is the main entry point for all users of the iMagine platform. If offers in a unified view: (1) a Marketplace with a catalogue of all modules available to iMagine users, developed both by iMagine users and by external communities, (2) a way to deploy those modules in cloud resources from iMagine consortium members (CSIC, INCD, TUBITAK) and beyond.
- **Inference platform** - https://inference-walton.cloud.imagine-ai.eu
  This is where iMagine users can deploy their services for production using OSCAR[3], leveraging the cloud resources of iMagine consortium members (Walton site).
- **FlowFuse instance** - https://forge.flows.dev.ai4eosc.eu
  This is where iMagine users can create complex workflows involving their AI modules.
- **MLflow server** - https://mlflow.cloud.imagine-ai.eu
  This is an MLflow instance deployed for iMagine users to allow them to log their experiments to optimise their model training.

[3] https://oscar.grycap.net/

- **AI4OS NextCloud** - https://share.services.ai4os.eu
  This is the service that allows data to be remotely stored and accessed from inside your deployment. Access is restricted to authenticated users.
- **GitHub** - https://github.com/ai4os-hub
  This is where the code of all the AI modules is stored. iMagine users have maintainer permissions in their own AI module repos.
- **AI4OS Template Hub** - https://templates.cloud.ai4eosc.eu
  This is where users can generate templates for the code of their AI modules, after filling a form.
- **DockerHub** - https://hub.docker.com/u/ai4oshub
  This is where the Docker images of the AI modules are stored.
- **CI/CD pipeline** - https://jenkins.services.ai4os.eu/job/AI4OS-hub
  This is where users can check that their AI modules have successfully passed all the quality checks enforced by the AI4OS stack.
- **Status of services** - https://status.ai4eosc.eu
  This is where users can check if a specific AI4OS service might be down for some reason.

Note that for the aquatic science community, the iMagine platform is hosting dedicated instances of some of the services listed above (e.g. the Dashboard[4], the MLflow instance[5], and the Inference Platform[6]), while others are common to all users of the AI4OS stack (e.g. the AI4OS Nextcloud storage). If any of the above endpoints get modified during the iMagine project, users will be notified and the documentation will be updated.

Please refer to the above links when those services are mentioned in the deliverable, unless stated otherwise. A complete list of links (including those less relevant to users) can be found here[7].

Registration

Using advanced AI4OS services, like training or storage, requires authentication. Accounts are created in the EGI Check-In[8]. To access iMagine resources, users have to apply for iMagine VO membership filling the EGI Check-In form. Once one's account is approved, the user will be able to access authenticated services like the Dashboard.

[4] https://dashboard.cloud.imagine-ai.eu/
[5] https://mlflow.cloud.imagine-ai.eu/
[6] https://inference-walton.cloud.imagine-ai.eu/
[7] https://docs.ai4eosc.eu/en/latest/others/other-links.html
[8] https://docs.ai4os.eu/en/latest/getting-started/register.html

User roles and workflows

The iMagine AI platform focuses on three different types of users, depending on what they want to achieve. These roles are extensively described in the documentation[9], and this guideline only provides a summary overview of them.

The basic user

This user would like to use AI modules that are already pre-trained and test them with their data, and therefore no machine learning knowledge is required. For example, these users can take an already trained module for phytoplankton species classification that has been containerised, and use it to classify their own images of phytoplankton species.

AI4OS can offer to this type of user:

- A catalogue full of ready-to-use AI modules to perform inference with their data;
- A GUI to easily interact with the inference service;
- Integrate the model with their own services deepaas API[10];
- Solutions to deploy in the Cloud or local resources;
- The ability to develop complex topologies by composing different modules, thanks to the REST API available in every module.

The intermediate user

The intermediate users want to retrain an available module to perform the same task but fine-tuning the model to their own data. They still do not need deep knowledge on modeling of machine learning model development, but typically do need basic programming skills to prepare their own data into the appropriate format, and some basic knowledge of Machine Learning. Nevertheless, they can re-use the knowledge being captured in a trained network and adjust the network to their problem at hand by re-training the network on their own dataset. An example could be a user who takes the generic image module for object detection/segmentation/classification like YoloV8 and retrains it for fish detection or cold-water coral segmentation.

AI4OS can offer to this user:

- The ability to train an out-of-the-box AI module from the catalog on user's specific dataset;
- Easily interact with the model using the deepaas API;
- Data storage resources to access their dataset (e.g. AI4OS-nextcloud);

[9] https://docs.ai4os.eu/en/latest/getting-started/user-roles.html
[10] https://docs.ai4os.eu/en/latest/reference/api.html

- A private instance of a Computer Vision Annotation Tool (CVAT) to annotate your dataset images;
- A private server to run Federated Learning trainings with Flower[11];
- The ability to use gpus to accelerate your training;
- Mlflow to track training experiments;
- Solutions to deploy the developed service on Cloud resources or locally;
- The ability to share the module with other users in the user's catalogue.

The advanced user

The advanced users are the ones who develop their own ML/DL models and, therefore, need to be competent in the area of AI/ML/DL. This would be the case, for example, if, apart from using the provided generic image modules, users want to perform an object localisation task, which is a fundamentally different problem. Therefore, they will design their own neural network architecture, potentially re-using parts of the code from other modules.

AI4OS offers to these users:

- A ready-to-use IDE (vscode, Jupyter) with the main DL frameworks as a containerised solution running on different types of hardware (cpus, gpus);
- Data storage resources to access their dataset (e.g. AI4OS-nextcloud);
- The ability to integrate experiment tracking with mlflow[12];
- Tutorials on performing different types of trainings (incremental learning, distributed learning);
- Solutions to deploy the developed service on Cloud resources or locally;
- The ability to share the module with other users in the open catalogue and provide all relevant metadata information;
- The possibility to integrate their AI module with the DEEPaaS API to enable easier user interaction or build further services on top of the AI module.

Typical scenarios

There are several typical scenarios depending on how users interact with the platform to develop AI-based image analytics services. For each of the roles/scenarios, this section describes how they are relevant for iMagine users.

[11] https://flower.ai/
[12] https://mlflow.org/

Try an existing module

In this case, users create a temporal deployment, useful for evaluating the model's functionalities[13]. Users can try models either by deploying them:

- In the platform (during a predefined limited amount of time). Models deployed in the platform will leverage a user-friendly GUI to visualise the inference inputs and outputs.
- Locally (personal computer, Cloud, Kubernetes, HPC, etc.) using our publicly available Docker containers.

The platform deployment option is open to any user, not only to members of the project, making it a great option to showcase the power of iMagine use cases to members of external communities.

Deploy an existing module for inference

In this case, users create a permanent deployment, more suitable to use in production services. There are several possibilities on how to make this deployment integrated into the iMagine Marketplace, each one with its own set of advantages and drawbacks[14], as shown in **Figure 1**:

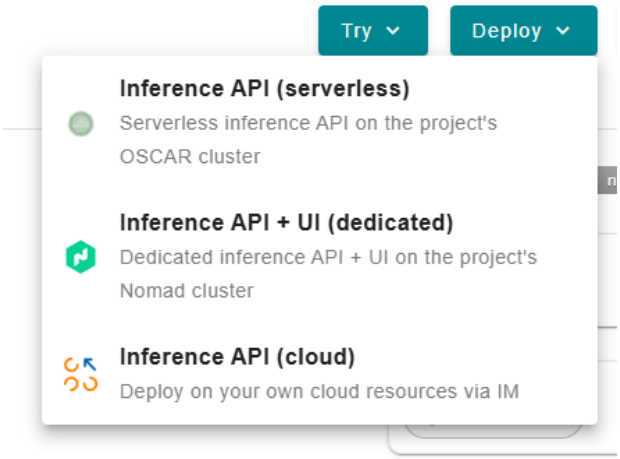

Figure 1 - Screenshot of the "Deploy" options in the iMagine Marketplace

- **Deploy in AI4OS (serverless)**: this is great if you want to consume resources only when you use the model, but will have some extra latency in each call;
- **Deploy in AI4OS (dedicated resources)**: this is optimal if you need minimal latency for inference calls, but will be consuming resources even when you are not actively using the model;

[13] https://docs.ai4os.eu/en/latest/howtos/try/index.html
[14] https://docs.ai4os.eu/en/latest/howtos/deploy/overview.html

- **Deploy in your Cloud resources**: this is excellent if you already have your own resources and want to deploy there, but it might need more effort to configure it properly.

In addition to the current inference options, the AI4OS stack is exploring the possibility of automatically deploying models in the EU node. In the following subsections, we provide a basic overview of the deployment workflows for each of these options.

Deploy in AI4OS (serverless)

To deploy an AI module in serverless mode[15] in the OSCAR iMagine cluster deployed at Walton[16]:

- Go to the Dashboard page of the AI module you want or deploy and select the serverless option;
- Go to the serverless deployment table and retrieve the information (ids, endpoints, etc.) Of your deployment;
- Following our tutorial script, use that information to either make synchronous or asynchronous predictions about your data.

All the use cases registered in the iMagine Marketplace have this deployment option available to offer their model for inference through OSCAR. This allows the models to process files in an event-driven approach. This is used, for example, in use case 1 (Litter assessment)[17] and use case 5 (Phytoplankton classifier)[18]. It is worth mentioning that, to satisfy the needs of the use cases, the OSCAR framework has evolved during the project. In the case of use case 1, OSCAR has added support for multi-tenancy[19], where multiple private buckets can be configured as the input for a single service, thus allowing the management of several users in the same OSCAR service.

Moreover, the use cases can also use other approaches supported by OSCAR to offer their model for inference. One of them is exposing the AI model's API or web interface using OSCAR exposed services: this approach is especially useful when the application executed provides its own API or user interface (e.g. Jupyter Notebook or Swagger interface). This approach avoids loading the AI weights each time the inference is triggered. This is the solution currently adopted by use case 2 (Zooscan[20]) to offer both the classifier and the separator model of their application, that are easily invoked by curl commands.

[15] https://docs.ai4eosc.eu/en/latest/howtos/deploy/oscar.html
[16] https://inference-walton.cloud.imagine-ai.eu/
[17] https://dashboard.cloud.imagine-ai.eu/marketplace/modules/litter-assessment
[18] https://dashboard.cloud.imagine-ai.eu/marketplace/modules/phyto-plankton-classification
[19] https://docs.oscar.grycap.net/multitenancy/
[20] https://dashboard.cloud.imagine-ai.eu/marketplace/modules/zooprocess-multiple-separator

We have also identified the need to process historical data, which typically consists of a significant number of files that need to be processed with the AI model. For that, we have developed during the project OSCAR-Batch[21], an open-source tool designed to perform batch-based processing using the OSCAR platform. It has a main component called the coordinator, which computes the optimal number of parallel service invocations for the state of the cluster and distributes the image processing workload accordingly. This ensures an efficient use of available CPU and memory resources in the OSCAR cluster, where multiple containers run in parallel. Each one loads the AI model weights just once and performs multiple inferences on a subset of the files. This approach is being used by use case 3o (OBSEA Fish Detection[22]) to process more than 1,000,000 images coming from historical data.

Deploy in AI4OS (dedicated)

To deploy an AI module with dedicated resources[23]:

- Go to the Dashboard page of the AI module you want or deploy and select the dedicated resources option;
- Select the resources you want to dedicate to your deployment (cpus, gpus, RAM);
- Go to the deployments table and retrieve the endpoints of your deployment
- Make predictions either using the Gradio UI endpoint, custom UI, or the deepaas API endpoint,

This deployment option is suitable if scaling of computing resources is not highly prioritised. It is also suitable for applications with a custom UI. In the iMagine project this is used, e.g. by the Oil Spill Detection use case (UC 4).

Deploy in your own cloud

To deploy an AI module in your own Cloud[24]:

- Go to the Dashboard page of the AI module you want or deploy and select the external Cloud option,
- You will be redirected to EGI Infrastructure Manager [25](IM), where you will be able to select the target Cloud, the number of resources you want to use, as well as additional configuration,

To further broaden the applicability and sustainability of AI modules, iMagine platform offers a way to deploy AI modules on the user's own cloud, given that it is supported and configured in IM. The supported clouds include EGI Cloud, AWS EC2, Microsoft Azure,

[21] https://github.com/grycap/oscar-batch
[22] https://dashboard.cloud.imagine-ai.eu/marketplace/modules/obsea-fish-detection
[23] https://docs.ai4eosc.eu/en/latest/howtos/deploy/oscar.html
[24] https://docs.ai4eosc.eu/en/latest/howtos/deploy/oscar.html
[25] https://www.egi.eu/service/infrastructure-manager/

Google Cloud, Openstack, EOSC EU Node VMs, and others. Because AI modules are provided as Docker images, one can deploy them on any platform supporting containers. Deployment on user custom resources is considered, e.g. by Litter Assessment, EMSO OBSEA, Oil Spill Detection, and other use cases.

Create an AI inference pipeline

In this case, users create a multi-stage inference pipeline to deploy their models[26]. They can stack different AI models and create modules to preprocess or post-process data generated at each step. Currently, the AI4OS stack supports creating pipelines with either Elyra or FlowFuse.

Deploy in Elyra

To create a pipeline with Elyra[27]:

- Create a Jupyter notebook in EGI notebooks;
- Clone the AI4OS repo with Elyra recipes;
- Using your EGI credentials, you will be able to access and connect the deployed serverless modules you have created in the Inference platform (cf. **Serverless deployments**);
- You can now start composing your pipeline in Elyra.

Deploy in FlowFuse

To create a pipeline with FlowFuse[28]:

- Create a new account on the flowfuse webpage;
- Create a new Node-Red instance;
- Connect the instance with the deployed serverless modules you have created in the Inference platform (cf. **Serverless deployments**);
- You can now start composing your pipeline in FlowFuse.

Use cases whose workflow application is composed of several AI models can benefit from the composition tools (like, for example, use case 2, composed of a separator model and a classifier model). However, none of the use cases have integrated their models yet with Elyra or FlowFuse, as they need to be first integrated with OSCAR (integration in some cases still under development or completed during the last months); still, we expect to have some examples before the end of the project.

[26] https://docs.ai4os.eu/en/latest/howtos/pipelines/index.html
[27] https://docs.ai4eosc.eu/en/latest/howtos/pipelines/elyra.html
[28] https://docs.ai4eosc.eu/en/latest/howtos/pipelines/flowfuse.html

Train an AI module

This scenario corresponds to an intermediate user who wants to train an existing module on their own dataset[29].

Although it is possible to train a module locally (since all the code is open source), the AI4OS platform focuses on allowing users to train models on cloud resources via the Dashboard. In that scenario, users should:

- Upload their dataset to AI4OS-Nextcloud storage;
- Deploy in the Dashboard the AI module they want to train. The deployment will be automatically connected to the storage;
- Retrain the module using the DEEPaaS API.

A canonical example of this scenario has been the usage of the AI4OS generic YOLO module to train many iMagine use cases, for example the OBSEA Fish Detection[30]. Other AI4OS modules that have proven to be useful for iMagine users are the generic image classifier[31] or the FastRCNN detection[32] module.

Optionally, before the training stage, users can profit from the AI4OS CVAT tool to accelerate their image labelling workflow with AI-powered annotations. This has been used for example by use case 7 to label images from beach video-monitoring systems, including a dataset for the beach seagrass wrack identification case (BWILD), and another one for detecting rip currents (currently in progress). BWILD and SCLabels are both publicly available in Zenodo.

Additionally, the AI4OS stack supports privacy-friendly Federated Learning trainings using Flower[33]. But due to the low sensitive nature of aquatic science data, this feature has not been especially relevant for iMagine users.

Develop an AI module

This scenario corresponds to advanced users that do not find an existing AI module according to their needs in the marketplace and thus want to develop a new one from scratch[34].

The workflow consists of:

[29] https://docs.ai4os.eu/en/latest/howtos/train/index.html
[30] https://dashboard.cloud.imagine-ai.eu/marketplace/modules/obsea-fish-detection
[31] https://dashboard.cloud.imagine-ai.eu/marketplace/modules/ai4os-image-classification-tf
[32] https://dashboard.cloud.imagine-ai.eu/marketplace/modules/ai4os-fasterrcnn-torch
[33] https://docs.ai4os.eu/en/latest/howtos/train/federated-server.html
[34] https://docs.ai4os.eu/en/latest/howtos/develop/index.html

- Using the AI4OS Module Template to create a skeleton of the module code with the predefined structure;
- Launching an interactive development environment to develop your module;
- Editing the model code and preparing the Dockerfile to deploy it;
- Create proper metadata for the model;
- Integrate the model in the AI4OS Marketplace[35].

During this development phase, users can profit from:

- Services like the dedicated mlflow instance to integrate experiment tracking in their modules;
- Tutorials on how to apply different types of learning paradigms (incremental learning, distributed learning).

The generic YOLOv8 module and the ZooProcess Multiple Classifier and Separator modules from use case 2 on the marketplace are examples of modules developed from scratch. For the YOLOv8 model, an advanced template was downloaded from the AI4EOSC Template Hub. The functions for prediction and training in the API script were then modified to integrate the DEEPaaS API into Ultralytics YOLOv8 model. Later, this module was used as a child module by several use cases, like OBSEA Fish Detection, to integrate the DEEPaaS API into the application.

Communication channels for users and providers

For the purposes of the iMagine project, support should be asked in the mailing list of the user work package "AI technical integration and support": imagine-wp3@mailman.egi.eu. This will enable iMagine users, i.e. researchers in aquatic sciences, to profit from each other's questions, which they may face themselves at some point. When questions might get too specific and are not covered by the general documentation of the platform, a dedicated meeting with the AI4OS support team can be organised on-demand. The email list above can also be used for collecting the feedback and ideas about the best practices and guidelines. The AI4OS services documentation is primarily hosted in GitHub[36] and any suggestions on improvements can be submitted via GitHub issues or direct contributions via Pull Requests.

[35] Keep in mind that the AI4OS platform role is to host the models, not to vouch for them meeting an arbitrary accuracy threshold. It is up to the user that shares the model with the community to provide details on the accuracy of the model.
[36] https://github.com/ai4os/ai4-docs

To further foster communication, there are planned dedicated webinars[37] and workshops[38] for users and providers. The iMagine organises regular Open Calls to attract use cases from the aquatic community and currently provides support for five external use cases, including collaboration with the DEAL project[39]. There is a general support email[40] for the border community as well as a dedicated email for external use cases[41].

Glossary

Please refer to the iMagine project-wide glossary[42].

[37] https://www.imagine-ai.eu/article/launching-the-imagine-webinar-series
[38] https://www.imagine-ai.eu/imagine-competence-centre
[39] https://pml.ac.uk/projects/deal-decentralised-learning-for-automated-image/
[40] imagine-ai-platform-support@mailman.egi.eu
[41] imagine-uc-oc@mailman.egi.eu
[42] https://confluence.egi.eu/display/IMPAIP/Glossary