

Date 11/05/2021
Version 2

**AI.Dielmo
Command line
Windows x86_64 Platform
USER MANUAL**

Prepared by:



José Luis Celda - Software Architect

Email: dielmo@dielmo.com
Skype: josecasrlosdielmo
Phone: +34 96 313 72 12

**Certificates ISO 9001 and ISO 14001 for
“Project development related with
cartographic services “**

INDEX

1.ABOUT AI.DIELMO COMMAND LINE VERSION	3
2.INSTALLATION	4
3.USAGE	4
3.1. Get help	4
3.2. Get help on the parameters of a task type	5
3.3. Interpreting the help parameters of a task	5
3.4. Queuing a task	6
3.5. List the queued tasks	7
3.6. BATCH CREATION FOR WINDOWS	8
3.7. Error codes	8
3.8. Diagnostic information	9
4. COMPONENTS AND DATA FLOW DIAGRAM	10

1. ABOUT AI.DIELMO Command line version

AI.dielmo, has been developed by DIELMO 3D with the aim of making more efficient the detection of objects in images by using Artificial Intelligence (AI), and perform actions on the detected objects, such as cropping the images with different parameters, selecting the best image among several with the same object, or simply counting the number of objects in the image.

The development of this software has been funded by the Spanish Ministry of Economic Affairs and Digital Transformation.

Call: Promotion of Digital Enabling Technologies.

Beneficiary: Dielmo 3D, S.L.

File no.: TSI- 100909-2019-47

Project: Analysis of the application of Deep Learning for the resolution of geospatial problems related to the territory.



AI.Dielmo is divided into two separate components:

1. The visual interface, (task controller, UI App).

It allows you to queue, stop, pause and consult the task history. (See manual for visual interface version).

2. The command line (task queueing, batchqueue)

The command line has been conceived to queue tasks to be performed by the UI App application. The task will remain queued indefinitely until the UI App is started and the enqueueing of tasks is activated (by default) (see manual visual interface version).

This manual develops the use of the second component: BatchQueue.

This component is designed to be used by any production flow, integrating object detection with the different tasks of a company.

The queued tasks are executed sequentially, by the queuing order. The order of the tasks in the queue can be altered using the UI App.

2. INSTALLATION

BatchQueue is automatically installed with the supplied installer for the selected platform (Windows/Linux). (See manual for visual interface version).

Note:

AI.Dielmo works with a JAVA virtual machine, which is packaged in the installer. If you try to use the JAVA virtual machine installed on Windows, and the version is not the appropriate one, the application will not start. To avoid this problem BatchQueue uses the appropriate virtual machine to invoke the application.

3. USAGE

The application is installed in `C:\Users\Public\AI.Dielmo`

In that folder, there is a command file with `.bat` extension with the name `batchqueue.bat` that is in charge of starting the application.

It can be invoked from any folder of the machine, you can put it in the `PATH` of the system if you want or start with the complete path. ex:

```
C:\Users\Public\AI.Dielmo\batchqueue
```

In the following descriptions, it will be assumed that it has been executed from the windows command window:

```
cd C:\Users\Public\AI.Dielmo
```

or the system path has been set to the route `C:\Users\Public\AI.Dielmo`

3.1. GET HELP

For general help on the use of the application use the following command:

```
c:\Users\Public\AI.Dielmo> batchqueue -help
```

The result of the command will be:

```
AI.Dielmo command line interface. batchqueue
```

```
Starting ApplicationQueue v0.0.1-SNAPSHOT using Java 11.0.9.1 on
YOURMACHINE with PID 11696
(C:\Users\Public\AI.Dielmo\AI.Dielmo.batch.jar started by jcelda
in c:\Users\Public\AI.Dielmo)
No active profile set, falling back to default profiles: default
Started ApplicationQueue in 3.517 seconds (JVM running for 5.336)
```

```
parameter jobName must be specified. Valid jobNames
values:[drawDetectionsJob, blurDetectionsJob, cropImagesJob,
drawDetectionsVideoJob, blurDetectionsVideoJob, vectorizeJob]
clue: batchqueue --help --jobName=drawDetectionsJob
Stopping job scheduler queue...
```

3.2. GET HELP ON THE PARAMETERS OF A TASK TYPE

For help on the parameters supported by a particular task type, use the following command:

```
batchqueue --help --jobName=<name of task>
```

ie: Ask for help on the task Mark detections on output images.

```
c:\Users\Public\AI.Dielmo> batchqueue -help -jobName=
drawDetectionsJob
AI.Dielmo command line interface. batchqueue
Starting ApplicationQueue v0.0.1-SNAPSHOT using Java 11.0.9.1 on
YOURCOMPUTER with PID 6068
(C:\Users\Public\AI.Dielmo\AI.Dielmo.batch.jar started by jcelda in
c:\Users\Public\AI.Dielmo)
No active profile set, falling back to default profiles: default
Started ApplicationQueue in 3.674 seconds (JVM running for 5.473)
drawDetectionsJob
Draw detections in image
A job to detect objects and generate output with images detected
```

Parameters:

```
--inputResources    Input Images Directory
No Default value defined
Input images Directory. This directory will be scanned for
supported images format to process detections.
Optional:false

--outputResources   Output Detections Directory
No Default value defined
The directory where the generated images and output results
summary file will be stored.
Optional:false

--tensorModel       Dielmo dmod model file
```

```
No Default value defined
A file provided by Dielmo with the trained neuronal network to
process detections. Extension should be .dmod
Optional:false
```

[...]

```
To launch a job with default values:
--inputResources=somevalue... --outputResources=somevalue...
--tensorModel=somevalue... --jobName=somevalue...
```

```
Stopping job scheduler queue...
```

3.2. INTERPRETING THE HELP OF TASK PARAMETERS

For each parameter, it is listed:

- Name (syntax) must be specified as listed.
- Short description.
- Default value (if any).
- Range of supported values (if any).
- Extended description.
- Optional: If a parameter is NOT optional and does NOT have a default value, it must be specified. In case it has a default value, it is not necessary to specify it, unless you want to overwrite the default value.

At the end of the help, the non-optional parameters with no default value are listed to have a reference on how to queue the task by specifying only these parameters, which are mandatory and have no default value.

3.3. QUEUING A TASK

3.3.1. CHECKING THE PARAMETERS

When launching the queuing command, some checks are performed first.

BatchQueue will check that all the parameters are correct, otherwise, it will list all the errors and the task will not be queued.

The checks that BatchQueue performs are:

All non-optional parameters have been specified and have a value.

The value of the parameters is correct if:

- Is within the allowed range of values.
- Is a directory or file name, which exists and is accessible.

An attempt to queue without specifying the required parameters is shown below as an example.

```
C:\Users\Public\AI.Dielmo>batchqueue --jobName=drawDetectionsJob
AI.Dielmo command line interface. batchqueue
Starting ApplicationQueue v0.0.1-SNAPSHOT using Java 11.0.9.1 on
JCELDAWIN with PID 8876
(C:\Users\Public\AI.Dielmo\AI.Dielmo.batch.jar started by jcelda in
C:\Users\Public\AI.Dielmo)
No active profile set, falling back to default profiles: default
Started ApplicationQueue in 3.517 seconds (JVM running for 5.325)
Error in parameters values:
Parameter [inputResources] cannot be null
Parameter [outputResources] cannot be null
Parameter [tensorModel] cannot be null
JOB NOT QUEUED
Stopping job scheduler queue...
```

3.3.2. SYNTAX FOR QUEUING A TASK

Use the following syntax:

```
C:\Users\Public\AI.Dielmo> batchqueue --jobName=drawDetectionsJob
--inputResources=Y:\temp\openai\person_images
--outputResources=Y:\temp\openai\output
--tensorModel=Y:\temp\openai\models\mask_rcnn_inception_v2_coco_201
8_01_28.dmod
AI.Dielmo command line interface. batchqueue
Starting ApplicationQueue v0.0.1-SNAPSHOT using Java 11.0.9.1 on
JCELDWIN with PID 12984
(C:\Users\Public\AI.Dielmo\AI.Dielmo.batch.jar started by jcelda in
C:\Users\Publi
c\AI.Dielmo)
No active profile set, falling back to default profiles: default
Started ApplicationQueue in 3.492 seconds (JVM running for 5.361)
Queing job:[drawDetectionsJob] with
parameters:[{inputResources=Y:\temp\openai\person_images,
outputResources=Y:\temp\openai\output,
tensorModel=Y:\temp\openai\models\mask_
rcnn_inception_v2_coco_2018_01_28.dmod, accuracy=50.0,
numberOfThreads=1, chunkSize=1, strokeSize=5, bufferSize=5,
bufferHeight=5, mergeType=Merge, intersectionPercent=70,
laplacianThreshold=0, useProcessorParts=false,
windowsSizeX=3600.0, windowsSizeY=4800.0,
overlapBetweenParts=1800.0, jobName=drawDetectionsJob}]
Job Queued...
Stopping job scheduler queue...
```

Note:

Directory and file names containing whitespace should be escaped using double quotes, as seen in the inputResources parameter.

```
C:\Users\Public\AI.Dielmo>batchqueue --jobName=drawDetectionsJob
--inputResources="Y:\temp\openai\person images"
--outputResources=Y:\temp\openai\output
--tensorModel=Y:\temp\openai\models\mask_rcnn_inception_v2_coco_201
8_01_28.dmod
```

3.4. LISTING THE QUEUED TASKS

To obtain a list of the queued tasks, the -list parameter is used, as shown in the following command:

```
C:\Users\Public\AI.Dielmo>batchqueue --list
AI.Dielmo command line interface. batchqueue
Starting ApplicationQueue v0.0.1-SNAPSHOT using Java 11.0.9.1 on
JCELDWIN with PID 7676
```



```
(C:\Users\Public\AI.Dielmo\AI.Dielmo.batch.jar started by jcelda in
C:\Users\Public\AI.Dielmo)
No active profile set, falling back to default profiles: default
Started ApplicationQueue in 3.531 seconds (JVM running for 5.323)
Queued Jobs:
Job Id:1619111236094 Queued since:2021 abr. 22 19:07:16 Job
Type:[drawDetectionsJob] Parameters:[#Thu Apr 22 19:07:16 CEST 2021
jobName=drawDetectionsJob
bufferHeight(long)=5
laplacianThreshold(long)=0
windowsSizeY(double)=4800.0
windowsSizeX(double)=3600.0
tensorModel=Y:\\temp\\openai\\models\\mask_rcnn_inception_v2_coco_
2018_01_28.dmod
accuracy(double)=50.0
outputResources=Y:\\temp\\openai\\output
overlapBetweenParts(double)=1800.0
mergeType=Merge
intersectionPercent(long)=70
strokeSize(long)=5
inputResources=Y:\\temp\\openai\\person_images
numberOfThreads(long)=1
chunkSize(long)=1
useProcessorParts=false
bufferWidth(long)=5
]
Stopping job scheduler queue...
```

3.5. BATCH CREATION FOR WINDOWS

You can create batch's to be able to queue many jobs using a batch script in windows.

In the installation directory of the application you can find a script which can be used for the enqueueing of different jobs to be done.

To do this we will create a batch file, in which we will add as many lines of calls to the batch queue as necessary:

We will create the file jobs.cmd

```
call batchqueue --jobName=drawDetectionsJob
--inputResources=Y:\temp\openai\person_images_1
--outputResources=Y:\temp\openai\output1
-tensorModel=Y:\temp\openai\models\mask_rcnn_inception_v2_coco_2018
_01_28.dmod
```

```
call batchqueue --jobName=drawDetectionsJob
--inputResources=Y:\temp\openai\person_images_2
--outputResources=Y:\temp\openai\output2
-tensorModel=Y:\temp\openai\models\mask_rcnn_inception_v2_coco_2018_01_28.dmod
call batchqueue --jobName=drawDetectionsJob
--inputResources=Y:\temp\openai\person_images_3
--outputResources=Y:\temp\openai\output3
-tensorModel=Y:\temp\openai\models\mask_rcnn_inception_v2_coco_2018_01_28.dmod
```

We save the file and we will be able to execute it by double-clicking on it or from a terminal window.

3.6. ERROR CODES

When BatchQueue is invoked, the process returns an execution result code in the variable %errorlevel%.

This result code allows knowing the final status of the command execution, useful for integration into the user's productive processes.

0	The command was executed correctly
1	Error in parameters (error details will be printed in the error output of the console)
2	Unexpected error (details of the error will be printed in the error output of the console)

The error code can be queried with the following command:

```
echo %errorlevel%
1
```

For example, launching the command with the inputResources directory parameter without escaping:

```
C:\Users\Public\AI.Dielmo>batchqueue --jobName=drawDetectionsJob
--inputResources=Y:\temp\openai\person_images
--outputResources=Y:\temp\openai\output
--tensorModel=Y:\temp\openai\models\mask_rcnn_inception_v2_coco_2018_01_28.dmod
AI.Dielmo command line interface. batchqueue
Starting ApplicationQueue v0.0.1-SNAPSHOT using Java 11.0.9.1 on
JCELDWIN with PID 14384
(C:\Users\Public\AI.Dielmo\AI.Dielmo.batch.jar started by jcelda in
C:\Users\Public\AI.Dielmo)
No active profile set, falling back to default profiles: default
```

```
Started ApplicationQueue in 3.489 seconds (JVM running for 5.203)
Error in parameters values:
parameter:[inputResources] [Y:\temp\openai\person] file does not
exists
JOB NOT QUEUED
Stopping job scheduler queue...
```

The output presents the descriptive error of the parameter (parameter:[inputResources] [Y:\temp\openai\person] file does not exist), and when printing %errorlevel% we obtain:

```
C:\Users\Public\AI.Dielmo>echo %errorlevel%
1
```

3.7. DIAGNOSTIC INFORMATION

In the directory C:\UsersPublicAI.DielmoLogs a text file is saved with all the outputs of CoreBatch executions.

It can be consulted for example using the following command:

```
C:\Users\Public\AI.Dielmo>more logs\ai-dielmo-queue.log
```

4. Diagram of components and data flow

