

# VISCERAL Benchmark 1

## Guidelines for Participation v1.2 (20131210)

### Document History

v1.2 - 20131210 - Update of Section 6.3 concerning the output xml file and the xml file row of the table of output files in Section 6.1

v1.1 - 20131108 - Section 7.2 on Landmark Localisation evaluation updated, added details about [participants@visceral.eu](mailto:participants@visceral.eu) mailing list

v1.0 - 20130808 - Released version of document

## 1. Introduction

### 1.1 Registration

The first step in participation is registration. This is done online on the following page:

<http://visceral.eu:8080/register/Registration.xhtml>

During the registration process, participants will be required to sign and upload a participation agreement.

Once the participant is registered, logging into the registration system will reveal the *participant dashboard*.

### 1.2 Virtual Machine

The medical imaging data is stored on the Microsoft Azure Cloud. When participants register successfully, they will receive a virtual machine (VM) in the Microsoft Azure cloud (Windows or Linux VMs are available), provided and financed by VISCERAL, with the support of Microsoft Research. The information for accessing the VM will appear on the participant dashboard in the registration system (after a delay of up to a week), along with a document containing further information on using the VM (“Azure Cloud Guidelines”).

Note that each participating group should only register for the VISCERAL Benchmark once. After successful registration, the person completing the registration will get root access to the assigned VM, and will be able to create logins for colleagues.

Please shut down the VM if it will not be used for a longer time.

### 1.3 Training Data

The training data can be accessed from the VM. It is highly recommended to also download the teaser data, which includes a tutorial — instructions for doing this are in the “Azure Cloud Guidelines” document (downloadable from the participant dashboard).

The *shared access key* for accessing the data is provided on the participant dashboard in the registration system once the VM is assigned. Please only access the data on the cloud from within the assigned VM — accessing the data from outside the cloud results in additional costs for the organisers. If you would like to have the data locally, access to an ftp server can be requested from the organisers.

A list of all files in the training data set can be downloaded from the participant dashboard in the registration system (“Visceral training set file list”).

## 2. Two Tasks of VISCERAL Benchmark 1

Participants can elect to participate in one or both of the following tasks within VISCERAL Benchmark 1:

1. Standard organ segmentation
2. Surprise organ segmentation

Benchmark 1 is described in more detail in VISCERAL Deliverable 4.1<sup>1</sup>. In particular, Section 3.1 describes the standard organ segmentation task, while the surprise organ segmentation task is described in Section 3.2.

Within these tasks, participants can elect to participate in one or more sub-tasks, summarised below and described in more detail in VISCERAL Deliverable 4.1. “Full run” segmentation tasks require that the organs be detected and then segmented, while “Half run” tasks are organ segmentation-only tasks, and do not require organ detection to be done. Participants wishing to participate in one of the “Half run” tasks should request access to the location centroid information from the organisers (send an e-mail to [Oscar.Jimenez@hevs.ch](mailto:Oscar.Jimenez@hevs.ch) with [ivan.eggel@hevs.ch](mailto:ivan.eggel@hevs.ch) in cc). This contains the location of the centroid of each of the organs to be segmented. Once access to the centroid information is granted, its shared access key will appear on the participant dashboard in the registration system.

### 2.1 Standard Organ Segmentation

The following sub-tasks are available for standard organ segmentation:

1. Full run segmentation
2. Full run landmarks
3. Half run segmentation

Participants can participate in a single sub-task, or both sub-tasks 1 and 2. Having access to the centroid information for sub-task 3 excludes participation in sub-tasks 1 and 2. It is possible for participants to change the sub-tasks in which they wish to participate during the training phase, until the submission deadline — please inform the organisers of any change. Programs of participants taking part in sub-tasks 1 and 2 will not have access to the centroid initialization data during the evaluation phase.

### 2.2 Surprise Organ Segmentation

The following sub-tasks are available for surprise organ segmentation:

1. Full run segmentation learner
2. Full run localization learner
3. Half run segmentation learner

Participants can participate in a single sub-task, or both sub-tasks 1 and 2. Having access to the centroid information for sub-task 3 excludes participation in sub-tasks 1 and 2. As for the standard organ segmentation (Section 2.1), it is possible to change the sub-tasks during the training phase by informing the organisers.

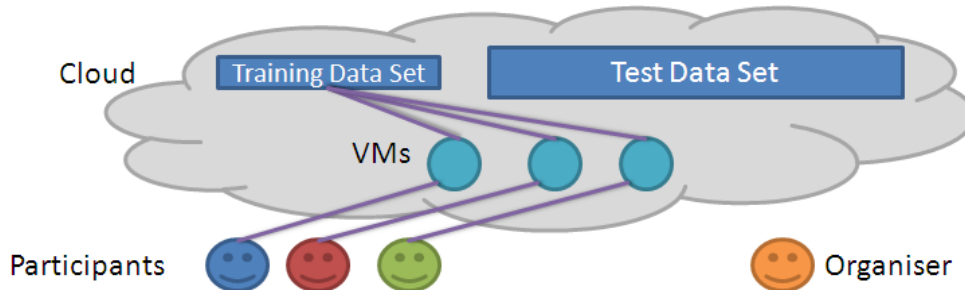
---

<sup>1</sup> VISCERAL Deliverable 4.1: <http://www.visceral.eu/assets/Uploads/Deliverables/VISCERAL-D4.1.pdf>

### 3. Benchmark Organisation

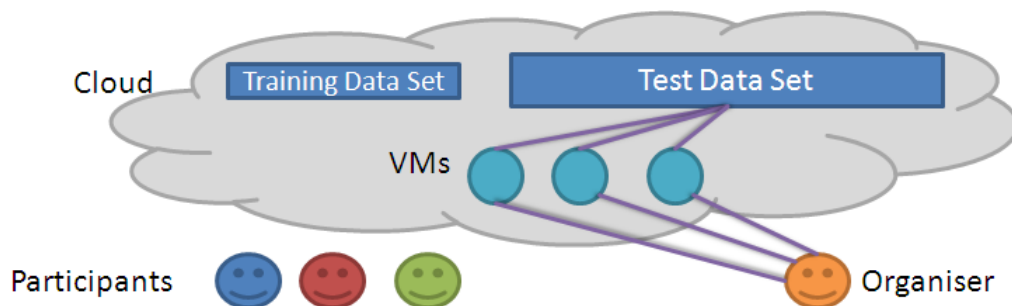
The benchmark runs in two phases.

#### 3.1 Training phase



The participants each have their own VM in the cloud, linked to a small annotated training data set of the same structure as the test data set. Software for carrying out the benchmark tasks must be placed into the VMs by the participants by the submission deadline. The software must be at least executable binaries and all libraries and other support required to execute the software. Source code is not required and must be removed from the VM by the submission deadline if the organizers should not see it (although even if source code is there, the organizers will not copy it to anywhere outside of the VM that it is in. Participants requiring additional security with respect to code or binaries can request to sign an NDA with the organizers). The software must satisfy all specifications in this document. The test data set is not accessible to the participants.

#### 3.2 Testing phase



On the benchmark submission deadline, the organizers take over the VMs from the participants, execute the software installed on the VMs on the unseen test data set and evaluate the results.

## 4. Further Information

Detailed information on the file formats used can be found in VISCERAL Deliverable 2.2.1<sup>2</sup>. Further information on the dataset and the modalities can be found in VISCERAL Deliverable 2.3.1<sup>3</sup>. This document also contains, in Section 3, the decisions taken regarding the manual annotation of the images. Participants will be provided with a list of all the names of files in the training data set.

All participants and organisers are automatically registered to the [participants@visceral.eu](mailto:participants@visceral.eu) mailing list, and can post on the mailing list. Use this list to communicate only among participants and the organisers, to ask questions, draw attention to problems or share hints and tips.

A LinkedIn group has been set-up for discussion about the Benchmark. Ask questions and make comments on this group:

<http://www.linkedin.com/groups/VISCERAL-Benchmark-Discussion-5089631>

You can also follow VISCERAL on Twitter: @VisceralEU.

## 5. Evaluation metrics for VISCERAL Benchmark 1

The following metrics have been selected for the evaluation in VISCERAL Benchmark 1:

### 5.1 Segmentation evaluation:

- Dice coefficient
- Average distance
- Adjusted Rand Index
- Interclass Correlation

### 5.2 Landmark Position Evaluation

- Landmark-specific average error (Euclidean distance) over all volumes, i.e. each landmark gets its own accuracy
- Percentage of detected landmarks per volume (i.e. landmarks detected / landmarks in the volume). This is a decision made by the participant. i.e., submitting 9 out of 18 landmarks in a volume means 50% detected.

---

<sup>2</sup> VISCERAL Deliverable 2.2.1:

<http://www.visceral.eu/assets/Uploads/Deliverables/VISCERAL-D2-2-1.pdf>

<sup>3</sup> VISCERAL Deliverable 2.3.1:

<http://www.visceral.eu/assets/Uploads/Deliverables/VISCERAL-D2.3.1.pdf>

## 6. Program Naming and Calling Conventions

After the submission deadline, the organisers will run the participants' programs installed in the VMs on the unseen test data. Please ensure that all of the following naming and calling conventions are followed, to ensure that this works smoothly.

### 6.1 Standard Organ Segmentation

One executable file (can be a script or compiled program) with the name

`execute_standard` or `execute_standard.{extension}`

must be in the home directory of the *azureuser* user (for those using Linux VMs), or on the Windows Desktop of the *azureuser* user (for those using Windows VMs) of the virtual machine. The *azureuser* user will be the only username available when the participant is assigned the virtual machine after registration.

#### Parameters

This executable file must take the following set of parameters, which are explained below:

```
-i [URL file to segment]
-o [output path]
-c [ConfigurationID]
-l [URL centroids for files to segment]
-r [RadLexIDs to segment]
-m
```

The image file URLs are constructed as:

`cURL+filename+saKey`

**cURL:** container URL,

`http://visceralstorage1.blob.core.windows.net/trainingset/`

**filename:** `PatientID_ModalityCounter_ModalityName_RegionName.nii.gz`

**saKey:** shared access key, e.g.

`?sr=c&si=readonly&sig=Z69O9Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D`

A full list of URLs of all the files in the training set can be downloaded from the registration system.

Standard organ segmentation	
Parameter	Explanation
-i [URL file to segment] <b>REQUIRED</b>	The URL of a file to segment + shared access key. The file will be in NIFTI.GZ format. The filename contains info on: PatientID, ModalityID, ModalityAbbreviation  <code>cURL+PatientID_ModalityCounter_ModalityName_RegionName.nii.gz+saKey</code>

	<p>Paths to a number of input volumes (In case of an MRT1 we might provide a corresponding MRT2 volume). The first input volume is the main volume that has to be segmented / landmarked. Additional volumes might be provided in case multiple modalities are available for this patient and region. Participants can (but don't have to) use these additional modalities if they have strong multimodal segmentation/localization algorithms. In general all volumes are original volumes, i.e., accurate registration has to be performed by the algorithm.</p>
<p>-o [output path] <b>REQUIRED</b></p>	<p>The full path in which output files are to be stored in:</p> <ul style="list-style-type: none"> <li>- Windows VM: temporary drive (D:)</li> <li>- Linux VM: /mnt/resource</li> </ul>
<p>-r [RadLexIDs to segment] <b>OPTIONAL</b></p>	<p>The program should segment the organs given by the sequence of one or more <b>RadLexID</b> taken from the following table:</p> <ul style="list-style-type: none"> <li>- Kidneys (two, most often) <ul style="list-style-type: none"> <li><i>Left Kidney (RadLexID 29663)</i></li> <li><i>Right Kidney (RadLexID 29662)</i></li> </ul> </li> <li>- Spleen (<i>RadLexID 86</i>)</li> <li>- Liver (<i>RadLexID 58</i>)</li> <li>- Lungs (two) <ul style="list-style-type: none"> <li><i>Left Lung (RadLexID 1326)</i></li> <li><i>Right Lung (RadLexID 1302)</i></li> </ul> </li> <li>- Urinary bladder (<i>RadLexID 237</i>)</li> <li>- Rectus muscle (two) <ul style="list-style-type: none"> <li><i>Muscle body of left rectus abdominis (RadLexID 40358)</i></li> <li><i>Muscle body of right rectus abdominis (RadLexID 40357)</i></li> </ul> </li> <li>- Lumbar Vertebra 1 (<i>RadLexID 29193</i>)</li> <li>- Thyroid (<i>RadLexID 7578</i>)</li> <li>- Pancreas (<i>RadLexID 170</i>)</li> <li>- Psoas muscle (two) <ul style="list-style-type: none"> <li><i>Left psoas major muscle (RadLexID 32249)</i></li> <li><i>Right psoas major muscle (RadLexID 32249)</i></li> </ul> </li> <li>- Gallbladder (without ductus) ( <i>RadLexID 187</i>)</li> <li>- Sternum (<i>RadLexID 2473</i>)</li> <li>- Aorta (<i>RadLexID 480</i>)</li> <li>- Trachea (<i>RadLexID 1247</i>)</li> <li>- Adrenal glands (two) <ul style="list-style-type: none"> <li><i>Left Adrenal Gland (RadLexID 30325)</i></li> <li><i>Right Adrenal Gland (RadLexID 30324)</i></li> </ul> </li> </ul>
<p>-m <b>OPTIONAL</b></p>	<p>The program should identify all landmarks present in the image</p>
<p>-l [URL location initializer centroids for files to segment] <b>OPTIONAL</b></p>	<p>Upon request only! This depends on the evaluation sub-task the participant is part of:</p> <ol style="list-style-type: none"> <li>1. fully automatic (sub-tasks 1 and 2), i.e., only image as input, the participant algorithm has to localize and segment from there</li> <li>2. initialization is provided (sub-task 3), i.e., we provide the centroids of organ segmentations as initializations</li> </ol> <p>The participant will have to indicate the choice at latest by the submission deadline. Participants participating in sub-task 3 will have access to a directory that holds the centroids of the organs that should be segmented.</p> <p><code>cURL+PatientID_ModalityCounter_ModalityName_RegionName_StructureInitializer.fcsv+saKey</code></p>
<p>-c [ConfigurationID] <b>REQUIRED</b></p>	<p>The program can have up to 5 configurations, indicated by the integers 1 to 5. A configuration can be for example different sets of segmentation parameters. The participants are free to choose what constitutes a configuration.</p>

## Organs to segment

It is not necessary that a program be able to segment all of the organs in the list. The program should accept all of the organ IDs in the above table, and provide no output for organ IDs that are not supported. Evaluation will only be done on the organ types that a program segments.

Similarly, if the program does not support landmark identification, it should accept the -m switch, but provide no output. Programs that do not do landmark identification will not be penalized in any way.

It is permitted to use different algorithms for different organs, but the executable file should take care of calling the correct algorithm for a given organ ID.

## Configuration IDs

The program should accept all five configuration IDs, and provide no output if a particular configuration ID is not implemented. During the evaluation phase, the organisers will begin with executing configuration ID 1, followed by the remaining configuration IDs in increasing order, until all configurations have been executed, or the period of weeks available for the evaluation calculations runs out. In the latter case, only the results for the configurations for which the execution completed will be provided.

## Output Files

Output	Explanation
segmentation files	<p>Segmentation file is one file per segmented organ. The output file must be written to the [output path]. The file is a NIFTI.GZ file that contains binary values:            0 ... background            1 ... organ of interest.            [0,1] ... fuzzy answers are possible, too, i.e. one value in the range 0 to 1 per voxel.</p> <p>The file name has to follow the convention:</p> <p><code>PatientID_ModalityCounter_ModalityName_RegionName_RadLexID_ParticipantID_ConfigurationID.nii.gz</code></p> <p>Note that the first part of the filename is identical to the volume file name the segmentation is performed on:  <code>PatientID_ModalityCounter_ModalityName_RegionName</code></p> <p>RadLexID ... identifies the anatomical structure            ParticipantID ... is a unique identifier for every participant (assigned by the registration system, visible on the participant information page after login).            ConfigurationID ... the same as the input parameter [configurationID]</p>
landmark files	<p>Files containing the coordinates of the localized landmarks. Formatting analogous to the annotation files in slicer3D convention, as specified in Section 6.4 of this document.</p> <p><code>PatientID_ModalityCounter_ModalityName_RegionName_Landmarks_ParticipantID_ConfigurationID.fcsv</code></p>
xml file accompanying every segmentation	<p>Must accompany a segmentation file only if further volumes beyond the volume to be segmented are used to obtain the segmentation (i.e. the volumes provided as second and further argument to -i). Same basis filename as the segmentation file:</p> <p><code>PatientID_ModalityCounter_ModalityName_RegionName_RadLexID_ParticipantID_ConfigurationID_info.xml</code></p> <p>The xml file holds the information specified in Section 6.3 of this document.</p>

## Example

**1. Segmentation output:** During the evaluation phase, the organisers will call the program of the participant with participantID **5xd9t** (who did not request to use location initialisers) as follows:

```
execute_standard -i
http://visceralstorage1.blob.core.windows.net/testset/CT1.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/MR1.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/MR2.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
-o /mnt/resource/output_p1/ -c 1 -r 1247 58
```

meaning that it should segment the trachea and liver in the file CT1.nii.gz using configuration 1, optionally using information from the files MR1.nii.gz and MR2.nii.gz and write the files:

```
/mnt/resource/output_p1/CT1_1247_5xd9t_1.nii.gz and
/mnt/resource/output_p1/CT1_1247_5xd9t_1_info.xml
and
```

```
/mnt/resource/output_p1/CT1_58_5xd9t_1.nii.gz and
/mnt/resource/output_p1/CT1_58_5xd9t_1_info.xml
```

The xml files should indicate if information from the additional MR1 and MR2 modalities was used, and if so, from which of these file the information was taken.

**2. Landmark output:** During the evaluation phase, the organisers will call the program of the participant with participantID **3pp58** as follows:

```
execute_standard -i
http://visceralstorage1.blob.core.windows.net/testset/001_1_CT_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D -o
/mnt/resource/output_p1/ -c 2 -m
```

meaning that it should produce all landmarks the algorithm can find in the image using configuration 2, and write the file:

```
/mnt/resource/output_p1/001_1_CT_wb_Landmarks_3pp58_2.fcsv
```

**3. Segmentation output making use of initialization centroids:** During the evaluation phase, the organisers will call the program of the participant with participantID **h23d8r** (who requested to use location initialisers) as follows:

```
execute_standard -i
http://visceralstorage1.blob.core.windows.net/testset/001_1_CT_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/001_1_MR1_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
-o /mnt/resource/output_p1/ -c 1 -r 1247 58 -l
001_1_CT_wb_StructureInitializer.fcsv
```

meaning that it should segment the trachea and liver in image 001\_1\_CT\_wb.nii.gz using configuration 1, optionally using information from image 001\_1\_MR1\_wb.nii.gz, and write the files:

```
/mnt/resource/output_p1/001_1_CT_wb_1247_h23d8r_1.nii.gz and
```



```
/mnt/resource/output_p1/001_1_CT_wb_1247_h23d8r_1_info.xml
```

and

```
/mnt/resource/output_p1/001_1_CT_wb_58_h23d8r_1.nii.gz
```

```
/mnt/resource/output_p1/001_1_CT_wb_58_h23d8r_1_info.xml
```

The xml files should indicate if information from the additional MR1 modality was used.

## 6.2 Surprise Organ Segmentation

One executable file (can be a script or compiled program) with the names

```
execute_surprise or execute_surprise.{extension}
```

must be in the home directory of the *azureuser* user (Linux) and on the Windows Desktop of the *azureuser* user (Windows) of the virtual machine. The *azureuser* user will be the only username available when the participant is assigned the virtual machine after registration.

### Parameters

This executable file must take the following set of parameters, which are explained below:

```
execute_surprise
```

```
-t [text file containing name list of training files and their annotations]
```

```
-i [text file containing name list of files to segment]
```

```
-o [output path]
```

```
-u [intermediate path for temporary files]
```

```
-c [configurationID]
```

```
-l [text file containing location initializer file names]
```

Surprise organ segmentation	
Parameter	Explanation
<pre>-t [text file containing name list of training files and their annotations]</pre> <p>REQUIRED</p>	<p>The text file is a list of full URLs to images and their ground truth segmentations forming the training set. It consists of pairs of URLs, with the first URL of each pair being the image data file, and the second of each pair being the ground truth segmentation, i.e.</p> <pre>cURL+PatientID_ModalityCounter_ModalityName_RegionName.nii.gz+saKey cURL+PatientID_ModalityCounter_ModalityName_RegionName_SurpriseOrganID.nii.gz+saKey</pre> <p>Image data files will be in NIFTI.GZ format. The filename contains info on: PatientID, ModalityID, ModalityAbbreviation, potentially region information</p> <p>Segmentation files contain the segmentation of a surprise organ for each of the training corpus image files. One file per surprise organ. The file is a NIFTI.GZ file that contains binary values: 0 ... background 1 ... organ of interest.</p>
<pre>-i [text file containing name list</pre>	<p>A text file containing the list of full URLs to files to segment. The file will be in NIFTI.GZ format.</p>

of files to segment] <b>REQUIRED</b>	The filename contains info on: PatientID, ModalityID, ModalityAbbreviation  cURL+PatientID_ModalityCounter_ModalityName_RegionName.nii.gz+saKey
-o [output path] <b>REQUIRED</b>	The full path in which output files are to be stored.
-u [intermediate path for temporary files] <b>REQUIRED</b>	Path where algorithms can store large intermediate files, eg. the learned segmentation model.
-l [text file containing location initializer file names] <b>OPTIONAL</b>	<p>Upon request only! This depends on the evaluation sub-task the participant is part of:</p> <ol style="list-style-type: none"> <li>1. fully automatic (sub-tasks 1 and 2), i.e., only image as input, the participant algorithm has to localize and segment from there</li> <li>2. initialization is provided (sub-task 3), i.e., we provide the centroids of organ segmentations as initializations</li> </ol> <p>The text file contains the list of full URLs to the location initialisation files, in the same order as the files in the text file containing the list of files to segment. Both lists should have the same number of filenames.</p> <p>The participant will have to indicate the choice at latest by the submission deadline. Participants participating in sub-task 3 will have access to a directory that holds the centroids of the organs that should be segmented.</p> <p>cURL+PatientID_ModalityCounter_ModalityName_RegionName_StructureInitializer.fcsv+saKey</p>
-c [configurationID] <b>REQUIRED</b>	The program can have up to 5 configurations, indicated by the integers 1 to 5. A configuration can be for example different sets of segmentation or machine learning algorithm parameters. The participants are free to choose what constitutes a configuration.

## Output Files

Output	Explanation
segmentation files (corresponding to files to segment)	<p>Segmentation file is one file per segmented organ. The file is a NIFTI.GZ file that contains binary values:</p> <p>0 ... background 1 ... organ of interest.</p> <p>The file name has to follow the convention: PatientID_ModalityCounter_ModalityName_RegionName_SurpriseOrganID_ParticipantID_ConfigurationID.nii.gz</p> <p>Note that the first part of the filename is identical to the volume file name the segmentation is performed on:</p> <p>PatientID_ModalityCounter_ModalityName_RegionName</p> <p>SurpriseOrganID ... identifies the anatomical structure ParticipantID ... is a unique identifier for every participant. ConfigurationID ... same as the input ConfigurationID parameter</p>

## Example

**No localisation:** During the evaluation phase, the organisers will call the program of the participant with participantID `5xd9t` as follows:

```
execute_surprise
-t trainingset.txt
-i files_to_segment.txt
-o /mnt/resource/output_p1/
-u /mount/temporary_p1/
-c 3
```

where the `trainingset.txt` is:

```
http://visceralstorage1.blob.core.windows.net/testset/001_1_CT_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/001_1_CT_wb_123.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/002_1_CT_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/002_1_CT_wb_123.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/003_1_CT_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/003_1_CT_wb_123.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
```

and `files_to_segment.txt` is:

```
http://visceralstorage1.blob.core.windows.net/testset/011_1_CT_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/012_1_CT_wb.nii.gz?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASpmpWZnT%2FhF2OgJOI7iEt60mis%3D
```

meaning that it should learn using configuration 3 from the images and ground truth segmentations (for RadLex ID 123) listed in `trainingset.txt`, and use the resulting classifier to classify the files in `files_to_segment.txt`. Output files should be:

```
/mnt/resource/output_p1/011_1_CT_wb_123_5xd9t_3.nii.gz
```

and

```
/mnt/resource/output_p1/012_1_CT_wb_123_5xd9t_3.nii.gz
```

**Localisation file used:** If the participant chooses to use localisation, then the option

`-l localisation_file.txt` should will appended to the call above, where

`localisation_file.txt` is:

```
http://visceralstorage1.blob.core.windows.net/testset/011_1_CT_wb_StructureInitialzer.fcsv?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASTmppZnT%2FhF2OgJOI7iEt60mis%3D
http://visceralstorage1.blob.core.windows.net/testset/012_1_CT_wb_StructureInitialzer.fcsv?sr=c&si=readonly&sig=Z6909Vz8TU0RxawtASTmppZnT%2FhF2OgJOI7iEt60mis%3D
```

## 6.3 Specification: xml file accompanying output

PatientID\_ModalityCounter\_ModalityName\_RegionName\_RadLexID\_ParticipantID\_ConfigurationID\_info.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<info>
<targetvolume>
  <patientid>PatientID</patientid>
  <modalitycounter>ModalityCounter</modalitycounter>
  <modality>ModalityName</modality>
  <region>RegionName</region>
</targetvolume>
<supportvolume>
  <patientid>PatientID</patientid>
  <modalitycounter>ModalityCounter</modalitycounter>
  <modality>ModalityName</modality>
  <region>RegionName</region>
</supportvolume>
<supportvolume>
  <patientid>PatientID</patientid>
  <modalitycounter>ModalityCounter</modalitycounter>
  <modality>ModalityName</modality>
  <region>RegionName</region>
</supportvolume>
</info>
```

### Example:

This file indicates that for the segmentation of the indicated MRT1 volume, the indicated MRT2 and CT volumes were used to obtain additional information to support the segmentation process. If only the target volume is used to obtain the segmentation, the support volume sections should be omitted.

```
<?xml version="1.0" encoding="UTF-8" ?>
<info>
<targetvolume>
  <patientid>000001</patientid>
  <modalitycounter>1</modalitycounter>
  <modality>MRT1</modality>
  <region>wb</region>
</targetvolume>
<supportvolume>
  <patientid>000001</patientid>
  <modalitycounter>2</modalitycounter>
  <modality>MRT2</modality>
  <region>wb</region>
</supportvolume>
<supportvolume>
  <patientid>000001</patientid>
  <modalitycounter>3</modalitycounter>
  <modality>CT</modality>
  <region>wb</region>
</supportvolume>
```

```
</info>
```

## 6.4 Specification: landmark file

```
PatientID_ModalityCounter_ModalityName_RegionName_Landmarks_ParticipantID_ConfigurationID.fcsv
```

```
#  
# Each row entry in this file contains a landmark specification for the respective  
# volume  
# A entry is structured as follows <landmark_identifier>, x-coordinate,  
# y-coordinate, z-coordinate  
# The coordinates are provided in world (scanner) coordinates  
#  
clavicle_right,x,y,z  
clavicle_left,x,y,z  
crista_iliaca_right,x,y,z  
crista_iliaca_left,x,y,z  
symphysis,x,y,z  
trochanter_major_left,x,y,z  
trochanter_major_right,x,y,z  
trochanter_minor_left,x,y,z  
trochanter_minor_right,x,y,z  
trachea_bifurcation,x,y,z  
aortic_arch,x,y,z  
aorta_bifurcation,x,y,z
```

## 6.5 Specification: segmentation centroid file

```
PatientID_ModalityCounter_ModalityName_RegionName_StructureInitializer.fcsv
```

```
#  
# Each row entry in this file contains a landmark specification for the respective  
# volume  
# A entry is structured as follows <radlexID>, x-coordinate, y-coordinate,  
# z-coordinate  
# The coordinates are provided in world (scanner) coordinates  
#  
RADLEXID,x,y,z  
RADLEXID,x,y,z  
RADLEXID,x,y,z  
RADLEXID,x,y,z  
RADLEXID,x,y,z  
RADLEXID,x,y,z  
RADLEXID,x,y,z  
RADLEXID,x,y,z
```

## 7. Evaluation Software

### 7.1 Segmentation Evaluation

To evaluate segmentations against the ground truth, the program

`EvaluateSegmentation` is provided. This software is available on the the virtual machine assigned to each participant.

#### Description

`EvaluateSegmentation` is a command that compares two volumes (a test segmentation and a ground truth segmentation) using 22 different metrics that were selected as a result of comprehensive research into the metrics used in the medical volume segmentations. `EvaluateSegmentation` provides the following measures:

#### Similarity

1. Dice Coefficient
2. Jaccard Coefficient
3. Area under ROC Curve (one system state)
4. Cohen Kappa
5. Rand Index
6. Adjusted Rand Index
7. Interclass Correlation
8. Volumetric Similarity Coefficient
9. Mutual Information

#### Distance

10. Hausdorff Distance
11. Average Distance
12. Mahanabolis Distance
13. Variation of Information
14. Global Consistency Error
15. Coefficient of Variation
16. Probabilistic Distance

#### Classic measures

17. Sensitivity (Recall, true positive rate)
18. Specificity (true negative rate)
19. Precision
20. F-Measure
21. Accuracy
22. Fallout (false positive rate)

#### Supported Images

`EvaluateSegmentation` supports all 3D file formats that are supported by ITK, e.g .nii, .mha, etc. The two Images should however have the same dimensions (the same number of voxels). There should be only one label in an image, where a voxel value can be either zero (background) or a value between zero and one  $[0,1]$  that denotes the fuzzy membership or

the probability that the corresponding voxel belongs to the label.

## Syntax

**USAGE:** EvaluateSegmentation truthURL segmentPath [-thd threshold]  
[-xml xmlpath] [-use all|DICE,JACRD, ...]

where:

truthURL: cURL+path to truth image+saKey

segmentPath: path to image being evaluated

-th threshold: before evaluation convert fuzzy images to binary using the given threshold

-xml xmlpath: path to xml file where results should be saved

-help: more information

-use metriclist: this option can be used to specify which metrics should be used. Metriclist consists of the codes of the desired metrics separated by commas.

For those metrics that accept parameters, it is possible to pass these parameters by writing them between two @ characters, e.g. -use MUTINF,FMEASR@0.5@. This option tells the tool to calculate the mutual information and the F-Measure at beta=0.5.

Possible codes for metriclist are:

all: calculate all available metrics (default)

DICE: calculate Dice Coefficient

JACRD: calculate Jaccard Coefficient

GCOERR: calculate Global Consistency Error

VOLSMTY: calculate Volumetric Similarity Coefficient

KAPPA: calculate Cohen Kappa

AUC: calculate Area under ROC Curve (one system state)

RNDIND: calculate Rand Index

ADJRIND: calculate Adjusted Rand Index

ICCORR: calculate Interclass Correlation

MUTINF: calculate Mutual Information

FALLOUT: calculate Fallout (false positive rate)

COEFVAR: calculate Coefficient of Variation

AVGDIST: calculate Average Distance

HDRFDST: calculate Hausdorff Distance HDRFDST@0.95@ means use 0.95 quantile

to

avoid outliers. Default is quantile of 1 which means exact Hausdorff distance

VARINFO: calculate Variation of Information

PROBDST: calculate Probabilistic Distance

MAHLNBS: calculate Mahanabolis Distance

SNSVTY: calculate Sensitivity (Recall, true positive rate)

SPCFTY: calculate Specificity (true negative rate)

PRCISION: calculate Precision

FMEASR: calculate F-Measure FMEASR@0.5@ means use 0.5 as a value for beta in the F-Measure

ACURCY: calculate Accuracy

## Examples

**Example 1:** `EvaluateSegmentation truth.nii segment.nii -use RNDIND,HDRFDST@0.96@,FMEASR@0.5@ -xml result.xml`

This example shows how to compare two NIFTI images providing the Rand Index, Hausdorff distance and the F-Measure and save the results in result.xml.

The values between two @ symbols are parameters to the specific measures in this case the quantile value used with Hausdorff distance to avoid outliers. The second value (0.5) is the beta value used with the F-Measure.

**Example 2:** `EvaluateSegmentation truth.nii segment.nii -use all -th 0.5`

This example compares two images using all available metrics. Before comparing the images, they are converted to binary images using a threshold of 0.5, that is voxels with values in  $[0,0.5)$  are considered as background and those with values in  $[0.5,1]$  are assigned the label with a membership of 1.

## 7.2 Landmark Localisation Evaluation

Landmark localization can be evaluated with the same Tool for evaluation of segmentations described in 7.1 using the option `-loc`. The files passed to the tool should fulfill the specifications described in 6.4.

USAGE:

`EvaluateSegmentation -loc truthLandmarkPath testLandmarkPath [-xml xmlOutputPath]`

where:

`truthLandmarkPath` = path (or URL) to file containing ground truth landmarks. URLs should be enclosed with quotations.

`testLandmarkPath` = path (or URL) to file containing landmarks to be evaluated. URLs should be enclosed with quotations.

`xmlOutputPath` = path to xml file where result should be saved.