MoCo GUI

v1.0

Contents

Documentation of GUIs for MR image registration

Thomas Küstner, Verena Neumann [*]

In this guide the usage of the GUIs for performing, evaluating and visualizing registrations of Magnetic Resonance (MR) images is documented. The developed GUIs are called

- RegGUI (1): for performing the registration
- EvalGUI (2): for registration visualization and quantitative evaluation
- LandmarkGUI (3): for evaluation via landmark points/lines/ROIs

1 RegGUI

RegGUI is a graphical framework to perform registration of time series of MR images with different registration algorithms.

1.1 Starting the GUI

Before the first start, you are asked to define some standard paths needed for proper GUI behavior (standard paths for input data and results). All path settings will be stored in a 'GUIPreferences.mat' file in the code directory. If there is no valid information provided, standard paths are set to your current working directory.

1.2 Usage

The standard layout of the GUI consists of a control panel and a field where the registration results can be displayed, see Figure 1.

1.2.1 Image registration

The supported registration algorithms are described in 4. Please download the respective packages before usage!

In order to perform an image registration, please follow these steps:
1. Input images can either be loaded from a file/directory ⊞ or directly from the Matlab workspace ⊞. Supported file formats are:

- DICOM
- NIfTI
- MAT
- MHD/RAW
- GIPL
- TIFF, JPG, BMP, PNG

Once images are loaded they are stored in the left list. Single-click on the image to view it in the below panel. Double-click on the image to change its geometry settings. Apply the changes by pressing the ✓ button, or abort the operation ⊘. Select image and press DEL-key to delete images from list.

2. Set registration parameters: registration method (see 4), 2D/3D image registration [1] and provide a parameter file. Check if appropriate output path is chosen.

3. Select two or more images (hold CTRL-key) which you want to co-register. Attention: Spatial image dimensionality must match!

4. Press ⊞ button and select the reference image onto which you are going to register all other selected images.

5. Images with selected registration algorithm are inserted into right list. Double-click on a registration to change its parametrization or press DEL-key to delete the selected registration.

6. Select one or more registration and press 'Run registration' to perform the image registrations.

[*] For support please contact Thomas Küstner: thomas.kuestner@iss.uni-stuttgart.de

[1] selecting 2D image registration of 3D input images results in looping over third dimension

This may take some time! Grab a coffee! When finished, the registration will turn green and is stored in a separate Matlab file in the output path. Single-click on it to get a preview of the registration result in the panel below.

1.2.2 Viewing images and deformation fields

Scroll through different slices via mouse wheel. If all four image spots are blocked, you need to clear the axes first by pressing the ☒ button on the right, before viewing new images/registration results.

1.2.3 Load already performed registrations

You may also restore already performed registrations by loading the generated Matlab result file 📂.

1.2.4 Evaluation

For further visualization and evaluation of the registration use the EvalGUI (2) or LandmarkGUI (3) application. It can be started with the selected registration results (registration must have been performed - green highlighted) from RegGUI.

EvalGUI is used for visualization of

- original and transformed images
- deformation fields
- divergence, determinant of Jacobian, gradient of deformation field, ...

and quantitative (metric-based) evaluation

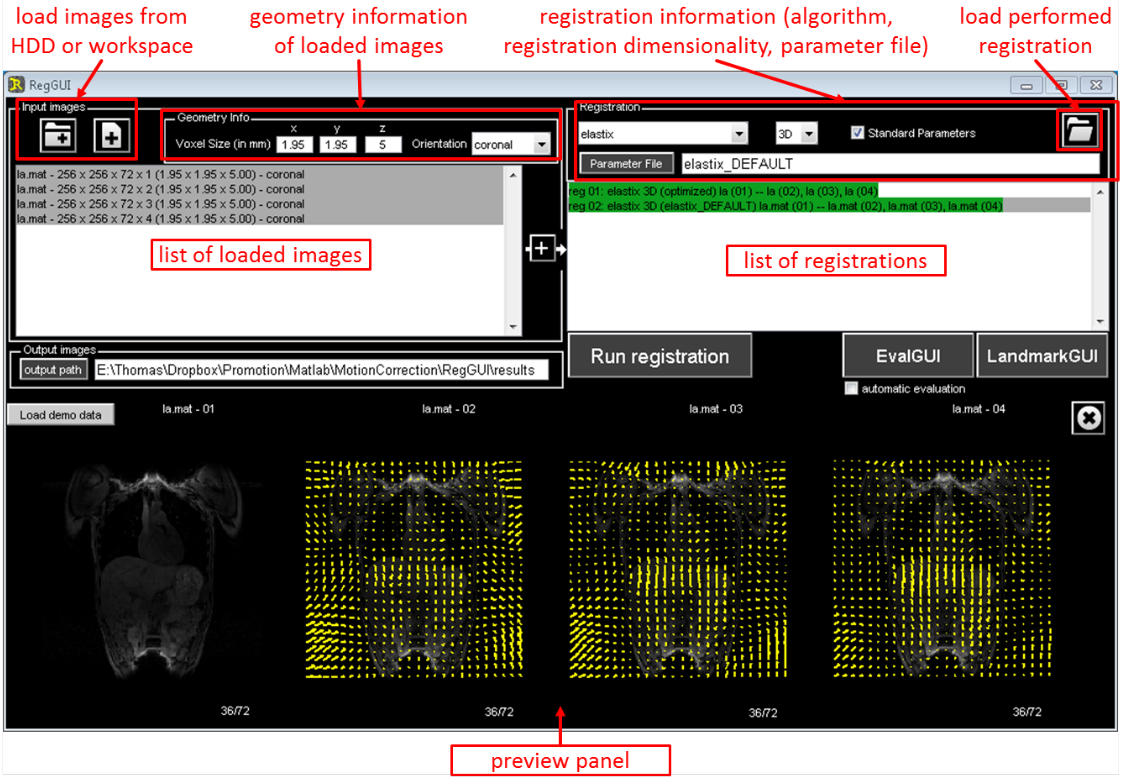
- automatic lung segmentation and measuring overlaps (Dice, Jaccard)
- intensity-based similarity metrics (SSD, MI, NMI, MSSIM, ...)

The intensity-based evaluation (EvalGUI) can also be performed automatically. If the check-box 'automatic evaluation' is checked the EvalGUI is called in automatic mode and computes overlap measures of the segmented lung and default similarity measures.

LandmarkGUI is used for feature-based evaluation via landmark points

- points
- lines
- ROIs

with derived overlap measures.



load images from HDD or workspace

geometry information of loaded images

registration information (algorithm, registration dimensionality, parameter file)

load performed registration

list of loaded images

list of registrations

preview panel

Voxel Size (in mm) x 1.95 y 1.95 z 5 Orientation coronal

la.mat - 256 x 256 x 72 x 1 (1.95 x 1.95 x 5.00) - coronal
la.mat - 256 x 256 x 72 x 2 (1.95 x 1.95 x 5.00) - coronal
la.mat - 256 x 256 x 72 x 3 (1.95 x 1.95 x 5.00) - coronal
la.mat - 256 x 256 x 72 x 4 (1.95 x 1.95 x 5.00) - coronal

output path E:\Thomas\Dropbox\Promotion\Matlab\MotionCorrection\RegGUI\results

elastix 3D ☑ Standard Parameters

Parameter File elastix_DEFAULT

reg 01: elastix 3D (optimized) la (01) -- la (02), la (03), la (04)
reg 02: elastix 3D (elastix_DEFAULT) la.mat (01) -- la.mat (02), la.mat (03), la.mat (04)

□ automatic evaluation

Figure 1: RegGUI: handling of registrations

1.2.5 Layout of registration result struct

This section describes the layout of the registration result struct which is needed to call the EvalGUI (2) and LandmarkGUI (3) from the command window and which is generated by the RegGUI (1). Bold variables are required, default values are shown in italic.

variable	description
dImg	2D/3D original images, dimensionality: x-y-z/slice-time; dImg(:,:,:,1) corresponds to the fixed/reference image and dImg(:,:,:,2:end) to the moving images
dImgReg	registered/transformed images, dimensionality: x-y-z/slice-time; dImgReg(:,:,:,1) corresponds to the fixed/reference image and dImgReg(:,:,:,2:end) to the transformed moving images
SDeform	$N \times 1$ struct containing forward (F) and backward (B) deformation fields (2D/3D) of each voxel and each direction; fieldnames: dBx, dBy, dBz, dFx, dFy, dFz
SGeo.cVoxelsize	$N \times 1$ cell array of double arrays containing voxel sizes of each image along x-y-z (*standardVoxelsize* in 'GUIPreferences.mat')
SGeo.cOrientation	$N \times 1$ cell array of strings (transverse|*coronal*|sagittal) for image orientation
iDim	registration dimensionality; 1=2D, 2=3D
nRegMethod	chosen registration algorithm needed for lung segmentation (select appropriate displacement calculation); *1=elastix*, 2=halar, 3=GRICS, 4=LAP, 5=demons
sRegMethod	string with registration algorithm (just needed for displaying purposes)
sShownames	$N \times 1$ cell array of strings with single filenames (just needed for displaying purposes)
sFilename	filename of fixed image (just needed for displaying and saving purposes)
sParFile	name of parameter file (just needed for displaying purposes)

Table 1: layout of registration struct

2 EvalGUI

EvalGUI is an application to visualize and evaluate registration results qualitatively and quantitatively. There are several ways to examine the results, e.g. display the registered images, the deformation field, the determinant of the Jacobian of the deformation field or the divergence, segment the lungs and check overlap measures. The lung segmentation requires thorax images.
You may either use the registration results obtained from RegGUI or load the GUI with your own (separately generated) registration results. For the latter case, please refer to section 1.2.5.

2.1 Starting the GUI

The GUI can be started from RegGUI with a selected (and performed) registration or from the command window. If you call it without input arguments, a dialog will appear to choose a dataset. Accepted command line arguments are:

- ('inarg', SRegiResult) : input struct containing registration result 'SRegiResult'
- ('path', sPath): path to Matlab file containing registration result
- ('egmode', true|false): switch auto evaluation mode on/off

All results are saved in the folder specified in the 'GUIPreferences.mat' file. If there is no 'SGeo' struct in your data, make sure the standard voxel size is defined correctly in 'GUIPreferences.mat'.

2.2 Usage

The GUI shows the first 4 gates/images and the corresponding deformation fields. On the left side there is a control panel, see Figure 2.

2.2.1 Viewing

Deformation field The size of the arrows of the deformation field can be changed and their distance can be adjusted. The scale and distance factor is displayed below the slider. For the size '1' the arrows are displayed in their actual length. The arrows can be removed by unchecking the box. One can switch between the forward and backward deformation field.

Images Different image views can be selected (coronal | sagittal | transverse) to display the images and the corresponding deformation fields in the respective projections. One can zoom into the images and scroll (via mouse wheel) through the different slices. The contrast and brightness can be adjusted by clicking and holding the central mouse button down and moving the mouse cursor (left-right for brightness, top-bottom for contrast) or alternatively with the left mouse button and `SHIFT`-key. The contrast can be reset by ↻.
If there are more than four temporal states in the dataset, the viewing of the other images can be selected via ≋.

Visualization The transformation $\vec{\Phi} = [\Phi_x, \Phi_y, \Phi_z]^T$ of the displacement field $\vec{F} = [F_x, F_y, F_z]^T$ is given as:

$$\vec{\Phi}(\vec{x}) = \vec{x} + \vec{F}(\vec{x}) \tag{1}$$

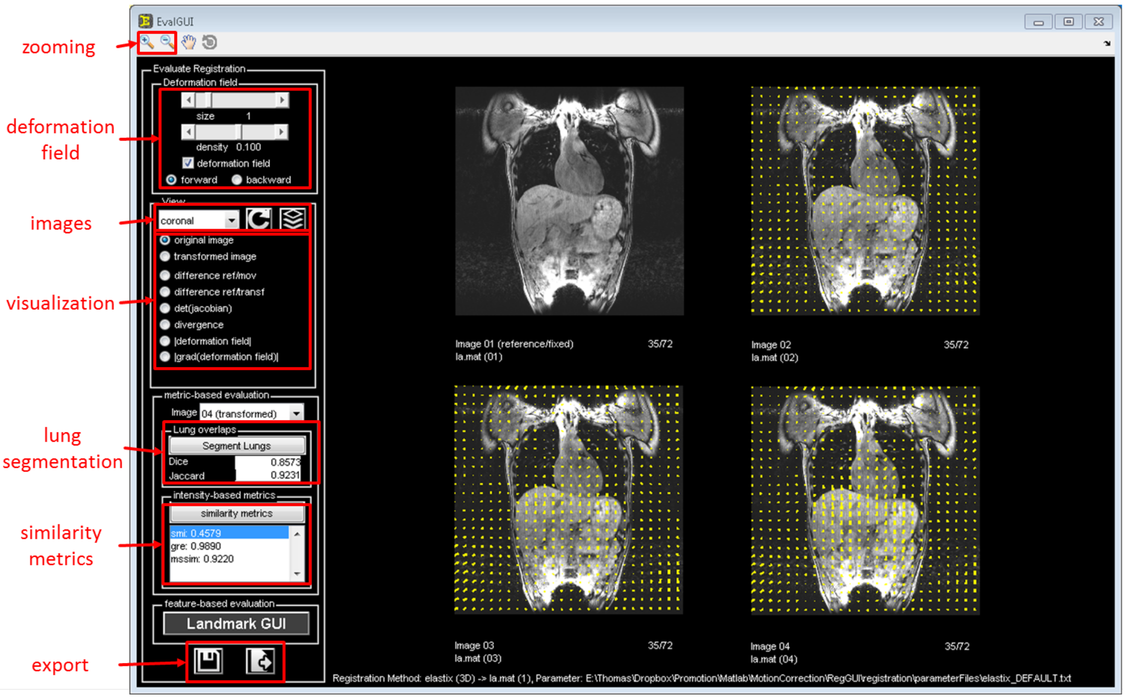
With the following definition of the nabla operator $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)$. All displacement field values are given in mm.
The radio buttons switch the viewing between

- the original images
- transformed/registered images
- difference between the original images (difference of moving to reference image is color-coded: red - no accordance (just reference image), green - no accordance (just moving image), yellow - perfect accordance)
- difference between the transformed images (difference of moving transformed to reference image is color-coded: red - no accordance (just reference image), green - no accordance (just moving image), yellow - perfect accordance)
- determinant of Jacobian

$$\begin{aligned}\mathrm{Jac}(\vec{\Phi}) = \det(\nabla\vec{\Phi}^T) &= \left|\begin{bmatrix} \frac{\partial\Phi_x}{\partial x} & \frac{\partial\Phi_x}{\partial y} & \frac{\partial\Phi_x}{\partial z} \\ \frac{\partial\Phi_y}{\partial x} & \frac{\partial\Phi_y}{\partial y} & \frac{\partial\Phi_y}{\partial z} \\ \frac{\partial\Phi_z}{\partial x} & \frac{\partial\Phi_z}{\partial y} & \frac{\partial\Phi_z}{\partial z} \end{bmatrix}\right| \\ &= \left|\begin{bmatrix} \frac{\partial F_x}{\partial x}+1 & \frac{\partial F_x}{\partial y} & \frac{\partial F_x}{\partial z} \\ \frac{\partial F_y}{\partial x} & \frac{\partial F_y}{\partial y}+1 & \frac{\partial F_y}{\partial z} \\ \frac{\partial F_z}{\partial x} & \frac{\partial F_z}{\partial y} & \frac{\partial F_z}{\partial z}+1 \end{bmatrix}\right|\end{aligned} \tag{2}$$

- divergence of transformation

$$\mathrm{div}(\vec{F}) = \nabla^T\vec{F} = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z} \tag{3}$$



size 1
density 0.100
☑ deformation field
● forward ○ backward

coronal
● original image
○ transformed image
○ difference ref/mov
○ difference ref/transf
○ det(jacobian)
○ divergence
○ |deformation field|
○ |grad(deformation field)|

Image 04 (transformed)
Dice 0.8573
Jaccard 0.9231

smi: 0.4579
gre: 0.9890
mssim: 0.9220

Image 01 (reference/fixed) 35/72
la.mat (01)

Image 02 35/72
la.mat (02)

Image 03 35/72
la.mat (03)

Image 04 35/72
la.mat (04)

Registration Method: elastix (3D) -> la.mat (1), Parameter: E:\Thomas\Dropbox\Promotion\Matlab\MotionCorrection\RegGUI\registration\parameterFiles\elastix_DEFAULT.txt

Figure 2: EvalGUI: visualization and quantitative evaluation

- | displacement field |

$$\triangle\vec{F} = |\vec{F}| = \sqrt{F_x^2 + F_y^2 + F_z^2} \tag{4}$$

- $|\nabla$ displacement field $|$

$$|\nabla\vec{F}| = \sqrt{\left(\frac{\partial F_x}{\partial p}\right)^2 + \left(\frac{\partial F_y}{\partial p}\right)^2 + \left(\frac{\partial F_z}{\partial p}\right)^2} \tag{5}$$

  Absolute displacement field change in direction $p \in \{x, y, z\}$.

2.2.2 Metric-based evaluation

Lung segmentation Overlap measures between the transformed moving and reference image can be calculated by lung segmentation for MR images which contain a thorax region. The segmentation can either be executed on the registered images or the original images. A region-growing algorithm (based on [1]) and consecutive Chan-Vese algorithm [2] segments the lungs. In case of segmenting the original images, the masks are transformed with the corresponding deformation fields before the comparison. The mask of the first temporal state is displayed in a figure, see Figure 3. The upper part can be cut off, because in the upper parts of the lobes the segmentation often fails. Hence, in order to derive reliable overlap measures (Dice and Jaccard), one can exclude the upper parts of the lung. The displayed values are in terms of the selected image to the reference image.

Similarity metrics Furthermore intensity-based metrics (which are often involved in image registration algorithms) can be calculated between the reference and the moving (transformed) images. By right-clicking on the button, one can select the to be determined similarity measures:
Shannon mutual information (SMI), Shannon normalized mutual information (SNMI), Renyi mutual information (RMI), Renyi normalized mutual information (RNMI), Tsallis mutual information (TMI), Tsallis normalized mutual information (TNMI), energy of joint probability (EJP), gradient entropy (GRE), F-information measures (FINFO), exclusive F-information (EFINFO), Structural similarity/Mean SSIM (SSIM/MSSIM), (Pearson) cross correlation (CC), zero-mean normalized (Pearson) cross correlation coefficient (ZCC), Spearman rank correlation (SPR), Kendall's tau (KET), mean squared error (MSE), normalized mean squared error (NMSE), root mean squared error (RMSE), normalized root mean squared error (NRMSE), peak

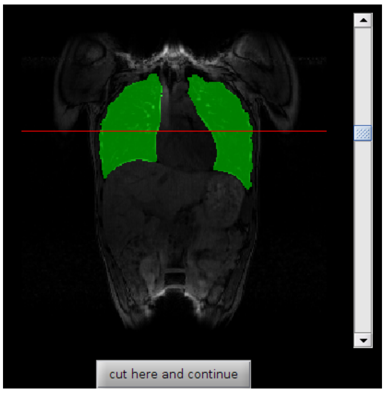


Figure 3: GUI to cut lung mask (for reliable overlap measures)

signal to noise ratio (PSNR), Tanimoto measure (TAM), zero crossings (absolute) (ZCA), zero crossings (relative) (ZCR), minimum ratio (MR), sum of squared differences (SSD), normalized sum of squared differences (NSSD), median of squared differences (MSD), sum of absolute differences (SAD), zero-mean sum of absolute differences (ZSAD), locally scaled sum of absolute differences (LSAD), median of absolute differences (MAD), sum of hamming distance (SHD), besov norm (BESOV)
which are after the calculation displayed in the box below for the selected image to the reference image.

2.2.3 Feature-based evaluation

For a feature-based evaluation of the registration results, one can examine anatomical landmark points in the LandmarkGUI (3).

2.2.4 Export

All obtained results can either be saved to a Matlab file 💾 or they can be exported ↪ to an Excel (*.xls,*.xlsx) or CSV (*.csv) file (left-click) or to the Matlab workspace (right-click) for further analysis.

3 LandmarkGUI

The LandmarkGUI is a tool to quantify the registration results manually. Landmarks (points, lines, ROIs) can be set in the reference image and registered images. For points, the Euclidean distance between the coordinates in the reference image and the registered image and for Lines and ROIS the Dice overlaps are quantitative measures for the registration accuracy.

3.1 Starting the GUI

The GUI can be started from the RegGUI (1) with the current dataset or from the command window without input arguments which will invoke a user interface to appear for choosing a dataset. The results are saved in the folder specified in the ’GUIPreferences.mat’ file. If there is no ’SGeo’ struct in your data, make sure the voxel size is defined correctly in the ’GUIPreferences.mat’ file or in the handed over registration result struct (1.2.5).

3.2 Usage

The GUI opens with the reference image on the left side and the second (moving) image on the right side. On the far left of the images is a control panel, see Figure 4.

3.2.1 View settings

The original or registered (transformed) images can be shown in coronal, sagittal and transverse orientation with the mouse wheel scrolling through the different slices. Contrast and brightness can be adjusted as described for EvalGUI (2) with the middle mouse button. For zooming and moving the image use the buttons in the toolbar.

3.2.2 Anatomical landmarks

First, decide if you want to mark the points in the original or registered images and choose the respective view. A Quick-Guide for labeling the anatomical landmarks is provided in ’LandmarkGUI-QuickGuide.pdf’.

Single points For a short synopsis see Figure 5.
1. To set single points, choose ’Points’ in the drop-down menu in the ’Set Points, Lines, ROIs’ Panel. Points can then be **set** in the images by clicking on the ’Set’ button or pressing `S` on the keyboard. The set function is applied to the image in focus (last updated) which is highlighted by a white frame. Use normal button clicks to add points. A `SHIFT`-, right-, or double-click adds a final point and ends the selection. Pressing `ENTER` ends the selection without adding a final point. Pressing `BACKSPACE` or `DELETE` removes the previously selected point. After finishing the set function all marked points are labeled from 1 to N (N being the number of set points) in yellow.

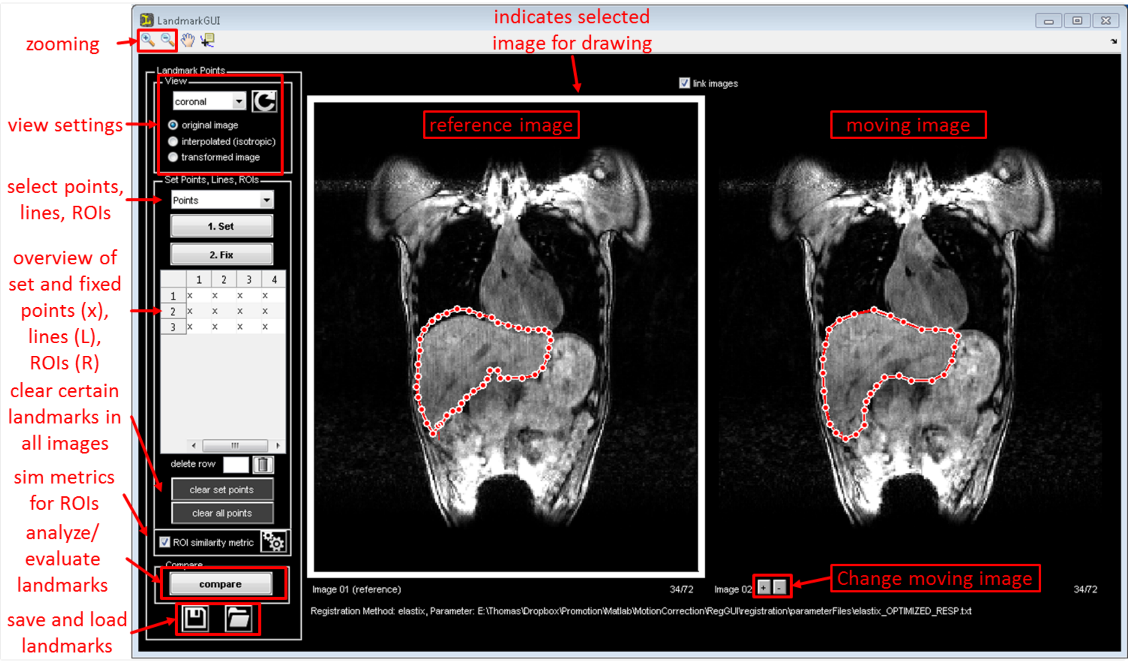
2. Set points in **all moving images**. You may change between the moving images with the ’+’ and ’-’ buttons below the moving image axes or with the `SPACE` key. All images should be labeled with the same number of points, in the same order so that corresponding numbers label the same landmark.

3. The set points can be **fixed** and finalized for comparison by clicking on the ’Fix’ button or pressing `F` on the keyboard. The marker and labels will turn red. A representation of all points is given in the table of the control panel. The columns contain the points of the different images (1 = reference image, 2 = moving image 1, etc.), corresponding points are located in the same row.

4. If you want to **change** the set points you can either delete whole rows, i.e. a fixed point in all images by indicating the row number in the text field below the table and clicking on 🗑 or you can move any point in any image by clicking the corresponding cell of the table. The respective label is then highlighted in blue and you can set it anew. The `ESC` key does not work for canceling the action but it is possible to set a point and delete it with `BACKSPACE` in order to stay with the previously set point. If you want to zoom into the image and improve the position in zoomed view make sure the point you delete is in the currently displayed image.

There are two more ways to delete points. You can clear the set points (’clear set points’) or delete all set and fixed points (’clear all points’) if you want to have a fresh start.

Lines and Regions of interest For a short synopsis of the labeling process see Figure 6.
1. To set lines or regions of interest select ’Lines’ or ’ROIs’ in the drop-down menu in the ’Set Points, Lines, ROIs’ Panel. Lines/ROIs can then be **set** in the images by clicking on the ’Set’ button or pressing `S` on the keyboard. The set function is applied to the image in focus (last updated) which is highlighted by a white frame. Use normal button clicks to add points. Double and right click finalize



View: coronal

● original image
○ interpolated (isotropic)
○ transformed image

Set Points, Lines, ROIs: Points

	1	2	3	4
1	x	x	x	x
2	x	x	x	x
3	x	x	x	x

☑ ROI similarity metric

☑ link images

Image 01 (reference) 34/72

Image 02 34/72

Registration Method: elastix, Parameter: E:\Thomas\Dropbox\Promotion\Matlab\MotionCorrection\RegGUI\registration\parameterFiles\elastix_OPTIMIZED_RESP.txt

Figure 4: LandmarkGUI: feature-based evaluation of landmarks

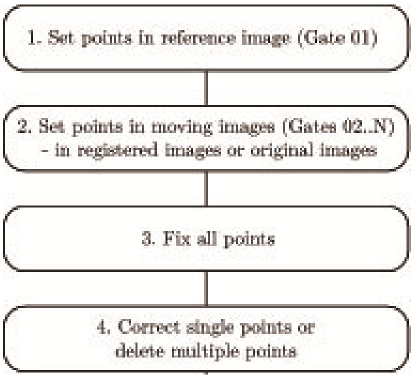

1. Set points in reference image (Gate 01)
2. Set points in moving images (Gates 02..N)
- in registered images or original images
3. Fix all points
4. Correct single points or
delete multiple points

Figure 5: Flow chart for the setting of single anatomical landmark points.

the line (double click sets a final point, right click does not). For regions of interest the final point is connected to the first point of the line so that a close region of interest is marked. You can now **edit** the vertex points of the line, if necessary. Drag the points, add new ones with `A` + mouse button click or delete points by selecting 'Delete Vertex' in the right click menu.

2. If the points are set correctly click on the 'Fix' button or press `F` on the keyboard to **fix** and finalize the lines. The lines and labels will turn red and can not be edited any more.

3. Now proceed with the labeling of the next image. Repeat steps 1. and 2. The lines and ROIs will be represented in the table in the control panel by L and R respectively for lines and ROIs in the images 1 to N. Label the same lines and ROIs in **all images**.

4. To **delete** lines and ROIs type the corresponding number in the text field and press 🗑. The line or ROI will be deleted in all images. Click on clear all lines/ROIs if you want to delete all set and fixed lines/ROIs.

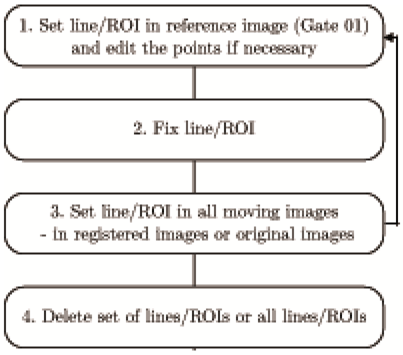

1. Set line/ROI in reference image (Gate 01) and edit the points if necessary
2. Fix line/ROI
3. Set line/ROI in all moving images - in registered images or original images
4. Delete set of lines/ROIs or all lines/ROIs

Figure 6: Flow chart for the setting of lines and regions of interest.

3.2.3 Comparison/Evaluation of landmarks

By pressing 'Compare' in the control panel, you invoke the evaluation of the landmarks. A new GUI opens with the evaluated results which also allows a saving, see Figure 7.

single points For each point $\vec{p} = [p_x, p_y, p_z]^T$ with number i the Euclidean distance d_i [mm] between the coordinates of the point in the moving image to the one in the reference image is evaluated. If points in the moving image were set on the transformed images, then it is $\forall\ 1 \leq i \leq M$

$$d_i = \sqrt{\left(\vec{p}_{\mathrm{ref}} - \vec{p}_{\mathrm{move}}\right)^T \mathbf{V} \left(\vec{p}_{\mathrm{ref}} - \vec{p}_{\mathrm{move}}\right)} \tag{6}$$

with the voxel spacing corrected matrix $\mathbf{V} = \mathrm{diag}\left(]v_x, v_y, v_z]\right)$. If points in the moving image were set on the original images, then it is $\forall\ 1 \leq i \leq M$

$$d_i = \sqrt{\left(\vec{p}_{\mathrm{ref}} - \vec{\tilde{\Phi}}\left(\vec{p}_{\mathrm{move}}\right)\right)^T \mathbf{V} \left(\vec{p}_{\mathrm{ref}} - \vec{\tilde{\Phi}}\left(\vec{p}_{\mathrm{move}}\right)\right)} \tag{7}$$

with in-plane projected movement $\vec{\tilde{\Phi}}\left(\vec{p}\right)$ of the point.
Afterwards the mean and standard deviation over all M point distances is presented for each moving image.

Lines and Regions of interest The lines and ROIs are evaluated by means of Dice overlap measure D. For the case of lines/ROIs, it is

$$D = \frac{2|m_{\mathrm{ref}} \cap \vec{\Phi}\left(m_{\mathrm{move}}\right)|}{|m_{\mathrm{ref}}| + |\vec{\Phi}\left(m_{\mathrm{move}}\right)|} \tag{8}$$

with masks m and $\vec{\Phi}$ descsribing the transformation if lines/ROIs were set on the original images and $\vec{\Phi} = \vec{x}$ (identity) if lines/ROIs were set on the transformed images.
Lines:
For the lines an additional Dice coefficient is evaluated, in which all pixels of the moving mask that are only 1 pixel unit away from the reference mask are counted. The given number is the ratio of the counted pixels to the number of the pixels that form the registered mask

$$D_{\mathrm{mod}} = \frac{2\left(N_{d,u} + N_{l,r}\right)}{|m_{\mathrm{ref}}| + |\vec{\Phi}\left(m_{\mathrm{move}}\right)|} \tag{9}$$

with the amount of overlapping pixels if the moving mask is additionally shifted by 1 pixel up/down $N_{d,u}$ or left/right $N_{l,r}$. The transformation $\vec{\Phi}$ is of course also applied if lines where placed in the original images. This yields then in total a Dice coefficient

$$D_{\mathrm{t}} = D + D_{\mathrm{mod}} \tag{10}$$

which measures the overlap of the moving to reference line and the by 1 pixel shifted moving line to reference line.
ROIs:
For ROIs one can additionally calculate (intensity-based) similarity metrics (see 2.2.2) of the regions.

3.2.4 Saving and loading of landmarks

All labeled landmarks can be saved to a mat-file, by clicking 💾. It is also possible to load a mat-file 🗀. Note that current fixed landmarks are deleted, when loading a new data set.

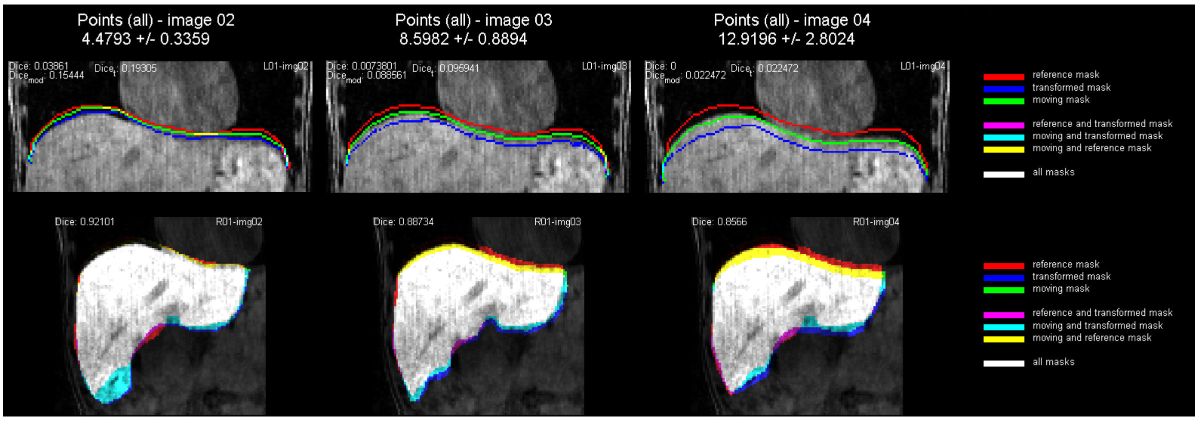


Figure 7: Comparison/Evaluation of landmarks with quantitative distance (points) and overlap measures (lines/ROIs).

4 Registration Algorithms

4.1 elastix

The function `fRegElastix` is a MATLAB wrapper that registers images to a reference image with elastix [3]. Elastix is a powerful wrapper to the ITK (Insight Segmentation and Registration Toolkit) for performing parametric registrations. Input arguments are the reference and moving images, the parameter file name, the registration dimensionality, a struct with geometric information (voxel size, orientation), a path name for the storage of the results and to the registration executable files. The function computes the deformation field and the registered images.

As elastix works with '.mhd' data files first all images (N = size(dImg,4)) are written to a corresponding file.

The registration is called by the system command 'elastix' with the the first image as reference image and the other $N - 1$ images as moving images. Subsequently, the 'transformix' command computes the deformation field that transforms the reference image to the moving image. Additionally, the backwards deformation field is computed.

The parameter file considered best for thorax MR image registration is supplied in `registration\parameterFiles` with `elastix_OPTIMIZED_RESP.txt`, an alternative is `elastix_DEFAULT.txt`

Prerequisites

Elastix is a free software. You can download it on http://elastix.isi.uu.nl/download.php.

4.2 HALAR/LREG

The function `fRegHalar` is a wrapper to call the hierarchical adaptive local affine registration ('halar' or 'lreg') algorithm from Buerger et al. [4]. It is a registration algorithm that splits the images hierarchically and registers variable-sized blocks on each level. Halar is a parametric registration based on B-Splines.

The input and output of `fRegHalar` are the same as for `fRegElastix`.

The algorithm works with gipl files, so the array data is written to the respective files.

The registration is called by the system command 'lreg'. The parameter file and the reference and moving image files are passed. After the registration also here a transformation command computes the deformation field between the images.

Prerequisites

The 'halar' algorithm is also freely available at http://www.isd.kcl.ac.uk/internal/hyperimage/.

4.3 LAP

The function `fRegLAP` is a wrapper for the Local All-Pass (LAP) algorithm which was developed by Gilliam and Blu [5, 6, 7, 8] and modified by Gilliam, Kuestner and Neumann [9]. The LAP is an optical flow-based method which reflects non-rigid deformations as local rigid displacements, i.e. local all-pass filtering operations. The code is mainly MATLAB based and kindly included in this toolbox package. Thanks to Chris Gilliam and Thierry Blu.

The input and output of fRegLAP are nearly the same as for fRegElastix, with the additional output of the moving and reference images, if an interpolation to an isotropic resolution has been performed.

The parameter file is a text file containing information about the interpolated resolution of the image (use 0 to stay with original anisotropic resolution), the pyramid levels defined by the filter size R (default: max:4 to min:0), the interpolation variant of the final transformation 'moving image + deformation field = approximated reference image' (choose between 'shiftedlinear' and 'cubicOMOMS') and the algorithm ('fast' or 'slow'). An example is:

```
dDeformRes = 0
SOptions.PyrMin = 2
SOptions.PyrMax = 4
SOptions.Interpolation='shiftedlinear'
SOptions.Algorithm = 'fast'
```

Further information

Further information can be found at https://sites.google.com/site/cwsgilliam/3D-LAP.

4.4 Demons

The function fRegDemons calls the Demons algorithm which is a diffusion-based registration method motivated by the optical flow equation [10]. The input and output of fRegDemons are the same as for fRegLAP.

The parameter file is a text file containing the registration parameters: the interpolated resolution of the image (use 0 to stay with original anisotropic resolution), parameters controlling the diffusion process, interpolation and registration method, and the used similarity measure. For more detailed information, please refer to the code.

```
dDeformRes = 2
SOptions.SigmaFluid = 8
SOptions.SigmaDiff = 1
SOptions.Interpolation = 'Linear'
SOptions.Alpha = 2
SOptions.Similarity = 'p'
SOptions.Registration = 'NonRigid'
SOptions.MaxRef = []
SOptions.Verbose = 1
```

Prerequisites

The Demons algorithm is freely available at http://www.mathworks.com/matlabcentral/fileexchange/21451-multimodality-non-rigid-demon-algorithm-image-registration

References

[1] P. Kohlmann, J. Strehlow, B. Jobst, S. Krass, J.-M. Kuhnigk, A. Anjorin, O. Sedlaczek, S. Ley, H.-U. Kauczor and M. O. Wielptz, "Automatic lung segmentation method for MRI-based lung perfusion studies of patients with chronic obstructive pulmonary disease," *International Journal of Computer Assisted Radiology and Surgery*, vol. 10, no. 4, pp. 403–417, Apr. 2015.

[2] T. F. Chan and L. Vese, "Active contours without edges," *Image processing, IEEE transactions on*, vol. 10, no. 2, pp. 266–277, 2001.

[3] S. Klein, M. Staring, K. Murphy, M. Viergever and J. Pluim, "elastix: A Toolbox for Intensity-Based Medical Image Registration," *IEEE Transactions on Medical Imaging*, vol. 29, no. 1, pp. 196–205, Jan. 2010.

[4] C. Buerger, T. Schaeffter and A. P. King, "Hierarchical adaptive local affine registration for fast and robust respiratory motion estimation," *Medical Image Analysis*, vol. 15, no. 4, pp. 551–564, Aug. 2011.

[5] P. Thevenaz, T. Blu and M. Unser, "Interpolation revisited [medical images application]," *IEEE Transactions on Medical Imaging*, vol. 19, no. 7, pp. 739–758, July 2000.

[6] T. Blu, P. Thevenaz and M. Unser, "Moms: maximal-order interpolation of minimal support," *IEEE Transactions on Image Processing*, vol. 10, no. 7, pp. 1069–1080, Jul 2001.

[7] T. Blu, P. Thevenaz and M. Unser, "Linear interpolation revitalized," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 710–719, May 2004.

[8] C. Gilliam and T. Blu, "Local All-Pass Filters for Optical Flow Estimation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Apr. 2015.

[9] C. Gilliam, T. Küstner and T. Blu, "3d motion flow estimation using local all-pass filters," in *Proceedings of the IEEE Symposium on Biomedical Imaging (ISBI)*, Prague, Apr. 2016.

[10] J. Thirion, "Image matching as a diffusion process: an analogy with maxwell's demons." *Medical image analysis*, vol. 2, no. 3, 1998.