Accompanying Material to
Automatic Motion Compensation of Free Breathing acquired Myocardial Perfusion Data by using Independent Component Analysis

Gert Wollny, Peter Kellman, Andrés Santos, and María-J. Ledesma-Carbayo

February 7, 2012

1 Implementation notes

1.1 Obtaining and compiling the code

The software for motion compensation of free breathing acquired myocardial perfusion data is implemented with in the MIA framework. MIA is a generic framework for gray scale image processing. Please follow the guidelines given in [1] in order to obtain the latest version of the source code and compile it.

The development of the library has continued since the published results were obtained. Nevertheless, the code of the software version related to this publication is tagged as *release-MIDAS*. To switch to this release and create a branch from it run

```
git checkout -b release-MIDAS icamotionstate
```

before running *cmake* and compiling the code. A basic introduction to the most important commands for using GIT [2] is given at [3]. The key source code files and their role in the implementation are outlined in Table 1.

1.2 Algorithmic description

The software implements ICA based motion compensation of free breathing acquired perfusion data sets (ICA-SP, ICA-T,ICA-T+SP), motion compensation using Pseudo-Ground-Truth estimation, motion compensation that exploits the quasi-periodicity of the free breathing motion (QUASI-P), and motion compensation that uses a serial registration scheme (SERIAL). All non-linear motion compensation implementations use the same image registration framework as back-end.

Before a registration is done, all images of a perfusion series need to be separated on a per slice basis and put into a work set. The program *mia-2dseries2sets* serves this purpose: it takes as input a series of DICOM files comprising the perfusion data and creates files segmentX.set, that hold a collection of the file names belonging to one slice in the temporal order of the perfusion series. The X stands for the slice number. Note that the slice and temporal information is taken from the acquisition data that needs to be present in the image files. Standard DICOM files usually provide this information.

Given such a work set with its image files, all registration methods first load this set and the image data. Then special images at the start of the series might be removed based on the skip command line parameter given. After this, the algorithms are run, and finally the resulting images and the workset are stored. If segmentation information was given in the workset file, this information will be updated according to the obtained registration.

1.2.1 ICA based approaches

Program: *mia-2dmyoica-nonrigid* for non-linear registration, and *mia-2dmyomilles* for linear registration.

Related source files:

- src: 2dmyoica-nonrigid.cc
- src: 2dmyomilles.cc
- mia/core: ica.(cc|hh), ica_template.(cxx|hh), slopestatistics.(cc|hh), waveletslopeclassifier.(cc|hh)
- mia/2d: perfusion.(cc|hh), ica.(cc|hh)

First, all image data is converted to floating point representation. Then, ICA is run for the first time.

Main programs	
src/2dmyoica-nonrigid.cc	ICA based motion compensation using non-linear registration
src/2dmyoperiodic-nonrigid.cc	motion compensation exploiting quasi-periodicity of free breathing
src/2dmyoserial-nonrigid.cc	motion compensation by using a serial registration procedure
src/2dmyomilles	ICA based motion compensation using only linear registration
src/2dmyopgt-nonrigid.cc	Pseudo ground truth based motion compensation scheme
ICA related source code	
mia/core/ica.(cc|hh)	ICA routines and interface to IT++
mia/core/ica_template.(cxx|.hh)	Templated implementation that allows to pass different types of data series to the core ICA routines.
mia/2d/ica.(cc|hh)	Specialization of the generic ICA class for 2D images
Code related to the analysis of the ICA mixing matrix	
mia/core/slopestatistics.(cc|hh)	A class to provide various information about a 1D series of data
mia/core/waveletslopeclassifier.(cc|hh)	A class that uses the slope statistics to classify a set mixing curves resulting from an ICA of a perfusion series
mia/2d/perfusion.(cc|hh)	ICA based analysis of 2D free breathing data – provides the means to create synthetic references as well as to segment the region of interest around the heart based on the ICA.
QUASI-P related source code	
mia/2d/similarity_profile.(cc|hh)	Evaluate similarity profiles of image series and a subset of a free breathing acquired series that belongs to the same breathing phase.
PGT related source code	
mia/2d/ground_truth_evaluator.(cc|hh)	PGT evaluation
mia/2d/groundtruthproblem.(cc|hh)	Optimization problem for the PGT estimation
Code related to non-linear registration	
mia/internal/nonrigidregister.(cxx|hh)	templated implementation of the non-linear registration algorithm
mia/2d/nonrigidregister.(cc|hh)	instantiation of the non-linear registration algorithm for 2D images
mia/2d/fullcost/image.(cc|hh)	implementation of a generic image based cost function
mia/2d/ppmatrix.(cxx|hh)	Class to support the fast evaluation of the divcurl regularization of spline based transformations
mia/internal/fullcost/divcurl.(cxx|hh)	generic interface to the divcurl regularization
mia/2d/fullcost/divcurl.(cc|hh)	instantiation of the divcurl regularization
mia/2d/cost/ssd.(cc|hh)	*Sum of Squared Differences* cost function kernel
mia/2d/cost/ngf.(cc|hh)	*Normalized Gradient Fields* cost function kernel
Code related to the minimization	
mia/core/minimizer.(cc|hh)	Base class for all minimizer plug-ins
mia/core/minimizer/gsl.(cc|hh)	Minimizers that stem from the GNU Scientific Library
mia/core/minimizer/nlopt.(cc|hh)	Minimizers that stem from the NLopt library
Code related to data set management and segmentations	
mia/2d/SegPoint.(cc|hh)	implementation of a point in the segmentation
mia/2d/SegSection.(cc|hh)	Class to represent a segmented section of the myocardium
mia/2d/SegStar.(cc|hh)	Class to represent the center and the circle best approximating the outer wall of the left ventricle myocardium.
mia/2d/SegFrame.(cc|hh)	class to represent one segmented frame of a perfusion series including the segmented sections and the star.
mia/2d/SegSet.(cc|hh)	Full perfusion series segmentation data set that consist of various frames
mia/2d/SegSetWithImages.(cc|hh)	Same as SegSet, but additionally it holds all the image data

Table 1: List of the key source files for the implementation of the motion compensation algorithms. Information about additional, generic classes that are used in the software can be obtained by using the documentation created from the source code by using *Doxygen*.

Running ICA To run ICA the data is prepared like follows:

1. copy image data to the Fast_ICA input matrix,
2. set the number of maximum ICA iterations (given on the command line),
3. set the number of components, if zero was given, run the following steps for six, five, and four components,
4. set non-linearity to FICA_NONLIN_TANH,
5. set the initial ICA approach. If your version of IT++ is patched regarding the bug [4], then the initial approach will be *deflation*, otherwise it will be *symmetric*.
6. run the ICA with the given number of components.
7. if the method was deflation and it failed, rerun the ICA with the symmetric approach.
8. depending on your command line flags, the mean is now stripped and the mixing matrix normalized
9. run the slope classifier on the obtained mixing matrix
10. select the number of components with the better separation

In the paper, the number of components was set to zero, i.e. four, five, and six components are tried and the best source separation selected, the IT++ library was fully patched so that the deflation approach was used first and mixing curve normalization was applied.

LV region of interest If requested and given a successful identification of the LV and RV enhancement cycle, a region of interest around the left heart ventricle is evaluated and extracted. Here, three possible approaches are supported: Extraction based on the difference image between the actual LV peak and RV peak enhancement, based on the difference between the LV and the RV feature images, and based on individual segmentation from the LV and RV feature images. The best approach seems to be the one based on individual segmentations that was also used in the paper. Given a successful ICA synthetic references can then be created and the registration can be run with the parameters given.

1.2.2 The QUASI-P approach

Program: *mia-2dmyoperiodic-nonrigid*
Related source files:

- mia/2d: similarity_profile.(cc|hh)
- src: 2dmyoperiodic-nonrigid.cc

1.2.3 SERIAL approach

Program: *mia-2dmyoserial-nonrigid*
Related source files:

- src: 2dmyoserial-nonrigid.cc

1.2.4 Pseude Ground Truth approach

Program: *mia-2dmyopgt-nonrigid*
Related source files:

- mia/2d: ground_truth_evaluator.(cc|hh), mia/2d/groundtruthproblem.(cc|hh)
- src: 2dmyopgt-nonrigid.cc

1.2.5 Non-linear registration

Non-linear registration is executed by minimizing the weighted sum of the image similarity measure and the regularization term. Before registration begins, the source and the reference image are normalized to have a common mean of zero and a deviation of one. The image metrics were applied without normalization, i.e. the value and the gradient was not divided by the number of image pixels and in all cases all pixels were used.

Related source files:

- mia/core: minimizer.(cc|hh),
- mia/core/minimizer: gsl.(cc|hh), nlopt.(cc|hh)
- mia/internal: nonrigidregister.(cxx|hh), divcurl.(cxx|hh), multicost.(cxx|hh)
- mia/2d: nonrigidregister.(cc|hh) ppmatrix.(cc|hh)
- mia/2d/fullcost: image.(cc|hh) divcurl.(cc|hh)
- mia/2d/cost: ssd.(cc|hh) ngf.(cc|hh)

2 Running a motion compensation algorithm

Given you have a directory *data* full of DICOM images that comprise a myocardial perfusion series, the following chain of commands will let you run the motion compensation algorithm using the ICA+SP method:

Since the motion compensation is run per slice, you need to separate the slices first and create the slice based sets. The following command will create files segmentX.set listing the files that belong to one slice of the myocardium with X being the number of the slice. The slices are separated based on the *IDSliceLocation* parameter of the DICOM files.

```
cd data
mkdir -p workdir
mia-2dseries2sets -o workdir *.dcm
```

To run the motion compensation by skipping the first two images on all slice series and using the optimal parameters as desribed in the papper you can now do run

```
cd workdir
for f in $(ls *.set); do
  mia-2dmyoica-nonrigid -k 2 -i $f -o reg$f \
                        -a 16 --c-rate-divider 2 \
                        -d 10000 --divcurl-divider 2 \
                        -O nlopt:opt=ld-var1,xtola=0.001,ftolr=0.001,maxiter=300
done
```

The registered images will be stored as reg*.dcm, and the according series will be named regsegmentX.set. Note, however, because of current limitations of teh DICOM support, the output files may not carry all information provided with the original DICOM images.

3 Significance matrices

Table 2: Significance of improvement of each method with respect to the others based on a paired one-sided t-test. A dark gray background indicates a high significance ($p < 0.05$) and a light gray background represents low significance ($p < 0.05$).

SERIAL versus ...

		QUASI-P		ICA-T		ICA-T-SP		ICA-T-PGT		ICA-SP	
	df	t	p	t	p	t	p	t	p	t	p
R^2	467	0.18	0.43	2.72	0.0034	-4.14	1.00	-2.95	1.00	-2.82	1.00
MNSE	467	2.72	1.00	0.39	0.65	6.72	1.00	4.59	1.00	3.41	1.00
σ	467	-1.38	0.084	-1.75	0.04	3.60	1.00	1.42	0.92	2.07	0.98
BRMSE	38	3.12	1.00	2.94	1.00	6.90	1.00	5.54	1.00	5.03	1.00
DICE	2261	-10.77	1.00	-10.66	1.00	-29.50	1.00	-23.51	1.00	-23.19	1.00

QUASI-P versus ...

		SERIAL		ICA-T		ICA-T-SP		ICA-T-PGT		ICA-SP	
	df	t	p	t	p	t	p	t	p	t	p
R^2	467	-0.18	0.57	2.32	0.01	-5.37	1.00	-3.04	1.00	-3.61	1.00
MNSE	467	-2.72	0.0034	-1.83	0.034	6.11	1.00	2.61	1.00	1.58	0.94
σ	467	1.38	0.92	-0.56	0.29	6.17	1.00	3.39	1.00	3.61	1.00
BRMSE	38	-3.12	0.0017	0.74	0.77	8.55	1.00	4.88	1.00	3.61	1.00
DICE	2261	10.77	2.2e-16	-1.49	0.93	-20.52	1.00	-13.48	1.00	-14.38	1.00

ICA-T versus ...

		SERIAL		QUASI-P		ICA-T-SP		ICA-T-PGT		ICA-SP	
	df	t	p	t	p	t	p	t	p	t	p
R^2	467	-2.72	1.00	-2.32	0.99	-7.19	1.00	-7.74	1.00	-5.07	1.00
MNSE	467	-0.39	0.35	1.83	0.97	6.87	1.00	6.97	1.00	2.57	0.99
σ	467	1.75	0.96	0.56	0.71	8.79	1.00	6.55	1.00	4.42	1.00
BRMSE	38	-2.94	0.0028	-0.74	0.23	3.74	1.00	3.46	1.00	1.89	0.97
DICE	2261	10.66	2.2e-16	1.49	0.068	-20.53	1.00	-16.34	1.00	-13.13	1.00

ICA-T-SP versus ...

		SERIAL		QUASI-P		ICA-T		ICA-T-PGT		ICA-SP	
	df	t	p	t	p	t	p	t	p	t	p
R^2	467	4.14	2e-05	5.37	6.3e-08	7.19	1.3e-12	3.10	0.001	1.42	0.078
MNSE	467	-6.72	2.7e-11	-6.11	1e-09	-6.87	1e-11	-4.29	1.1e-05	-2.81	0.0026
σ	467	-3.60	0.00018	-6.17	7.6e-10	-8.79	2.2e-16	-4.99	4.3e-07	-1.37	0.085
BRMSE	38	-6.90	1.7e-08	-8.55	1.1e-10	-3.74	0.00031	-3.67	0.00037	-2.04	0.024
DICE	2261	29.50	2.2e-16	20.52	2.2e-16	20.53	2.2e-16	14.53	2.2e-16	7.82	4.1e-15

ICA-T-PGT versus ...

		SERIAL		QUASI-P		ICA-T		ICA-T-SP		ICA-SP	
	df	t	p	t	p	t	p	t	p	t	p
R^2	467	2.95	0.0017	3.04	0.0012	7.74	3.1e-14	-3.10	1.00	-0.80	0.79
MNSE	467	-4.59	2.8e-06	-2.61	0.0046	-6.97	5.3e-12	4.29	1.00	-0.42	0.34
σ	467	-1.42	0.078	-3.39	0.00038	-6.55	7.7e-11	4.99	1.00	1.16	0.88
BRMSE	38	-5.54	1.2e-06	-4.88	9.5e-06	-3.46	0.00068	3.67	1.00	-0.31	0.38
DICE	2261	23.51	2.2e-16	13.48	2.2e-16	16.34	2.2e-16	-14.53	1.00	-2.20	0.99

ICA-SP versus ...

		SERIAL		QUASI-P		ICA-T		ICA-T-SP		ICA-T-PGT	
	df	t	p	t	p	t	p	t	p	t	p
R^2	467	2.82	0.0025	3.61	0.00017	5.07	2.9e-07	-1.42	0.92	0.80	0.21
MNSE	467	-3.41	0.00036	-1.58	0.057	-2.57	0.0052	2.81	1.00	0.42	0.66
σ	467	-2.07	0.019	-3.61	0.00017	-4.42	6.1e-06	1.37	0.91	-1.16	0.12
BRMSE	38	-5.03	6.1e-06	-3.61	0.00044	-1.89	0.033	2.04	0.98	0.31	0.62
DICE	2261	23.19	2.2e-16	14.38	2.2e-16	13.13	2.2e-16	-7.82	1.00	2.20	0.014

References

[1] https://sourceforge.net/apps/mediawiki/mia/index.php?title=Installation

[2] http://git-scm.com/

[3] http://schacon.github.com/git/everyday.html

[4] https://sourceforge.net/tracker/?func=detail&aid=3028968&group_id=37044&atid=418758