SIEMENS

Applications Guide

Advanced EPI for fMRI

(Gradient-reversal, PLACE, Physio-Logging)

&

EPI 2D Distortion Correction

(Works-in-Progress)

MAGNETOM Avanto, Espree, TaTS Systems

syngo MR Numaris 4 VB15A

Feb 2008

WIP # 383

_____

Important Note

This document provides a description of techniques developed by Siemens. Siemens has tested the software provided with this works-in-progress package in combination with the proposed clinical application. However, each user should be aware of the fact that incorrect use of this software may produce unknown results.

The sequences contained in this software package do not exceed the FDA safety performance parameter guidelines for MRI exams. Specifically, there is no change to patient risk as compared to routine operation of the MAGNETOM with regard to: static magnetic field; the time rate of change of the gradient magnetic fields; the rate at which RF power is deposited into the body (SAR); or the acoustic noise created by the MAGNETOM.

> The software has been tested internally but not yet in a clinical environment. For routine applications, its functionality may not be complete, and use of this software will remain investigational.
>
> In general, the clinical user will, in its sole responsibility, decide on the use of this application package or on subsequent therapeutic or diagnostic techniques and shall apply such techniques in its sole responsibility.
>
> Siemens will not take responsibility for the correct application of, or consequences arising from use of, this applications package.

The software in this package may change in the future, or may not be available in future software versions. Siemens has the right to remove this software at any point. In case of any questions that are related to the use of this package please contact one of the WIP authors listed on the next page.

Authors

Dr. Josef Pfeuffer
Siemens Medical Solutions USA
MR RD Neuro Collaborations
josef.pfeuffer@siemens.com

PLACE original C source code was provided by
> Q-S. Xiang, Vancouver and F. Q. Ye, Bethesda.

Prototyping contributions for the EPI Distortion correction were provided by
> Kawin Setsompop, MIT, Boston
> Susan Sporer, Theoretical Physics, U of Erlangen

The EPI 2D Distortion correction (PLACE, Gradient-reversal) was improved and implemented in C++ within the ICE framework by
> Markus Vogler, Fachhochschule Hof during his internship and master thesis.

Physio logging was improved and implemented by
> Vibhas Deshpande, Siemens Medical Solutions USA

Table of Contents

1. Description of the package

This application package consists of slightly changed sequences for fMRI. The following features are implemented, interfaced on a second exam UI tab cards:

Updated feature: a comprehensive ***General Linear Model (GLM) with advanced functionality (flexible design matrix: multiple conditions & multiple contrasts, arbitrary time base and amplitude)***. This allows a flexible analysis also of more complicated paradigms; paradigm definition is now done with a single ASCII input file in INI format.

This application package consists of slightly changed sequences for fMRI. The following features are implemented, interfaced on a second exam UI tab cards:

New feature: ***EPI 2D Distortion Correction*** based on two different encoding schemes and algorithms:

Gradient-reversal (Parameter: EPI PE direction) and

PLACE (Parameter: PLACE phase encoding, PLACE shift).

New feature: ***Physio logging*** which records the PMU signals in a log file.

2. Gradient reversal - Inversion of the EPI phase-encode direction

Echo planar imaging is intrinsically prone to off-resonance effects due to its long read-out time. Dependent on the local magnetic field, localization errors occur and distort the image in the direction of phase encoding. The distortion can be modified by time-reversal. Inversion of the read-out is only selectable for a single-shot EPI sequences (i.e. Segments = 1).
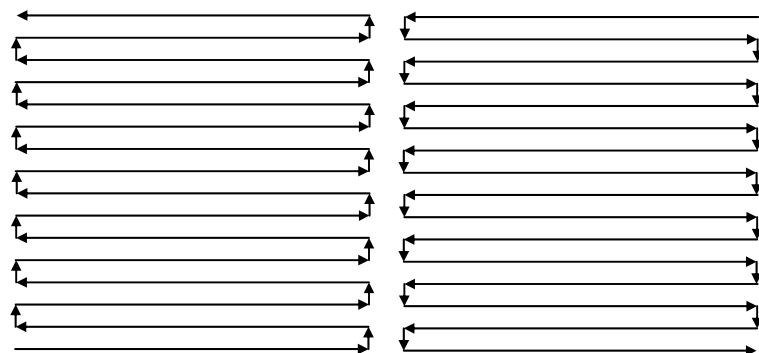
k-space coverage of an EPI sequence using different read-out directions.

In case of a reversed trajectory an off-resonance effect results in the same accumulation of a linear phase over time as before but with a different sign. As shown in the example below.
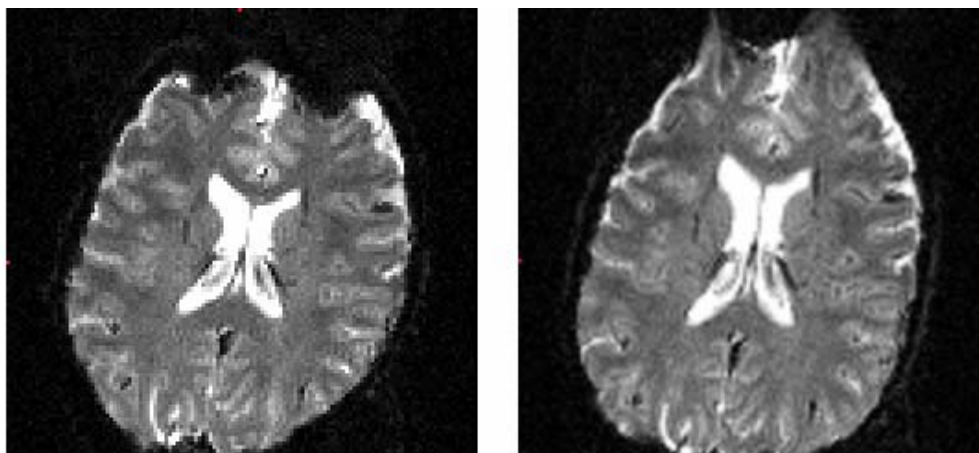
Example data of a single shot EPI acquisition with different read-out directions.
Left: As normal product. Right: reversed.

User Interface

Toggle "EPI PE direction" on the sequence special card (see chapter 1).

This WIP can also be used to correct susceptibility artifacts with a method called Gradient Reversal (H. Chang, J. M. Fitzpatrick IEEE Trans. Med. Imaging, 11(3):319-329, 1992). This method uses two images with different phase encoding directions (one compressed and one expanded) to compute a corrected image (see chapter 4).
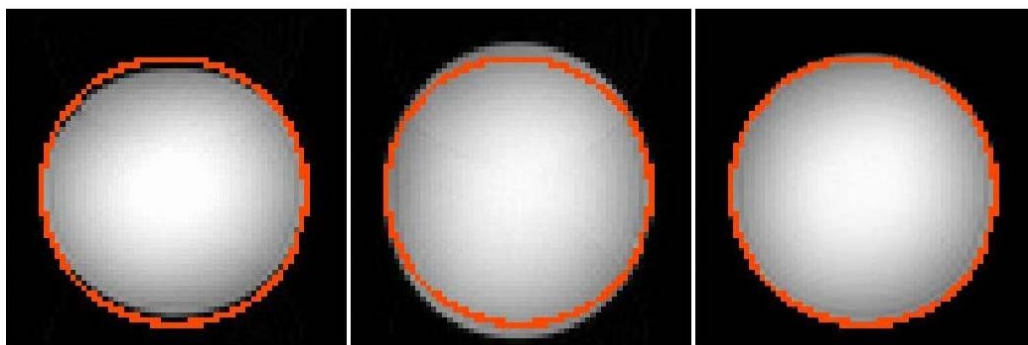
Figure: Compressed image (left), expanded image (middle) and corrected image (right)

3. PLACE shift for the EPI phase-encode direction

Echo-Planar-Imaging (EPI) suffers from two major artifacts:

    (i) N/2 ghost caused by differences between odd and even echoes;

    (ii) geometric/ intensity distortions due to signal misregistration along phase encoding (PE) direction.

The current method uses Phase Labeling for Additional Coordinate Encoding, **PLACE** (Q-S. Xiang, F. Q. Ye, ISMRM 2006, pg. 2346, Magn Reson Med. 2007, 57:731-41) to correct for either artifact. Using a PLACE shift, the pre-phase gradient in PE direction is changed by a small amount (in units of "blips"), which induces a phase change in the complex image.
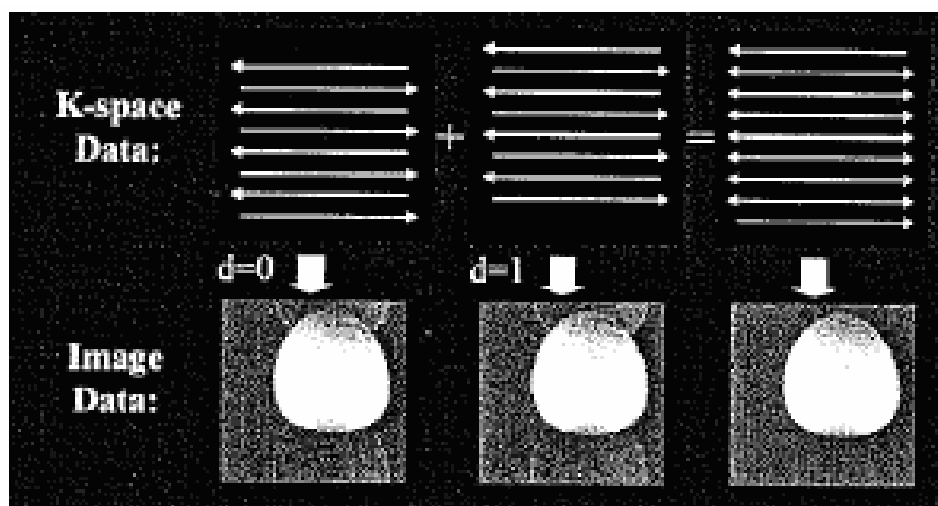
(1) For N/2 deghosting only. A PLACE shift of d = 1 creates a phase ramp within (-π,π) range across the entire field-of-view (FOV). This corresponds to a phase rotation of π radians for any N/2 ghost components relative to the overlapping image component since they are from two voxels located FOV/2 apart. By applying a phase ramp within (-π,π) range across the FOV, the two complex images can be essentially "phase aligned" and simply averaged to produce an image with highly suppressed N/2 ghost.

(2) For undistortion only. If the N/2 ghost is not present, the data of (1) can also be used for distortion correction, since the correct y-coordinate for each pixel in the undistorted space is encoded in the phase difference between the two complex images as, y = 0.5 FOV Arg (I1 I2*)/π. To achieve intensity restoration, sub-pixel mapping and 2D smoothing results in a displacement map for continuous warping.

(3) For undistortion in the presence of N/2 ghost. Here, a PLACE shift of d = 2 is used, which makes PLACE insensitive to the N/2 ghost, since the N/2 ghost will rotate by nearly 2π relative to the overlapping image component. Unique warping can be achieved as long as no voxel is displaced by more than FOV/4.

(4) For both N/2 deghosting and undistortion. Three images I1 , I2 , and I3 are acquired with PLACE shifts of d = 0, 1, 2. Two deghosted images can be obtained first by combining I1 and I2, as well as I2 and I3 using method (1). The two deghosted images can then be combined for undistortion using method (2).
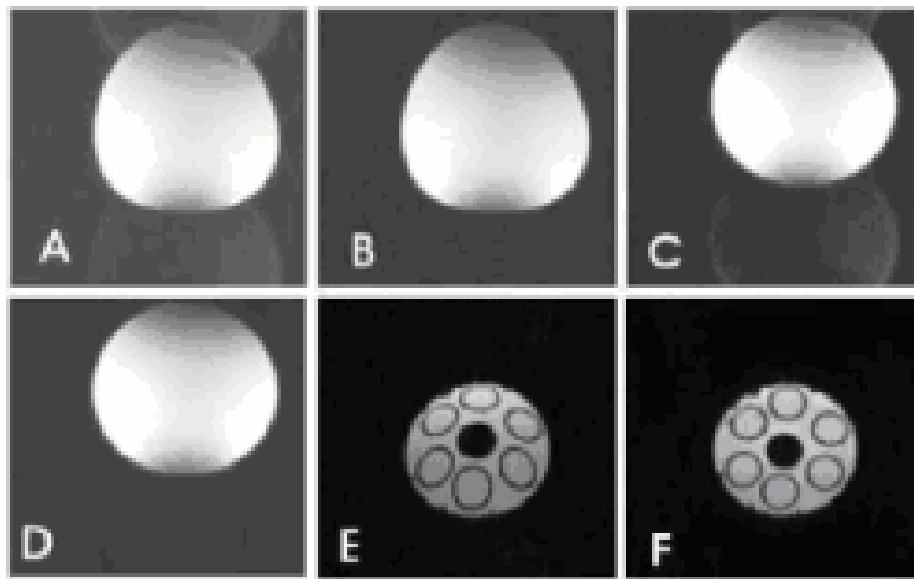
Figures (A-D) are images from a spherical phantom. A is one of the acquired images with both N/2 ghost and distortion; B is only deghosted with 2 images differing (d = 0, 1); C is only undistorted with 2 images (d = 0, 2); D is both deghosted and undistorted with 3 images (d = 0, 1, 2). E and F are distorted and corrected images from a phantom containing circular tubes.

User Interface

Set the PLACE shift d (in units of "blips") for the Phase Encoding direction on the sequence special card (see chapter 1). Typically are the combinations:

PLACE shift: shift in k-space, which is applied in all repetitions, default: 0

PLACE phase encoding: different dynamic encoding schemes can be chosen

Constant: Every repetition is acquired with the same PLACE shift

-1,+1: Odd repetitions are acquired with an shift of -1, even repetitions with a shift of +1. A non-zero *PLACE shift* will be added to all repetitions.

-1,0,+1: Three different shifts are applied alternating with each repetition.

A combination of *PLACE shift* (e.g. +1) and *PLACE phase encoding* (e.g. -1, 0,+1) can be used to create a different dynamic encoding scheme (e.g. 0, 1, 2)

How to store image data in complex format

Menue: Options -> Service -> Local Service -> TestTools -> RHP -> Rawdata On: Non-image DICOM data are created, which contain the complex k-space data. Simple 2D FT creates the complex images.

4. EPI 2D Distortion correction

EPI distortion correction can be enabled on the "Sequence – Special" card (screenshot in chapter 1): EPI Dist2D – On / Off. Two possible methods are available:

Gradient Reversal uses magnitude information to correct the miss map of the voxels caused by B0 inhomogenities.

PLACE uses phase information to calculate a displacement map to correct the miss map of the voxels caused by B0 inhomogenities.

2 complex images are needed to correct geometric distortions.

3 complex images are needed to correct geometric distortions and N/2 ghosting.
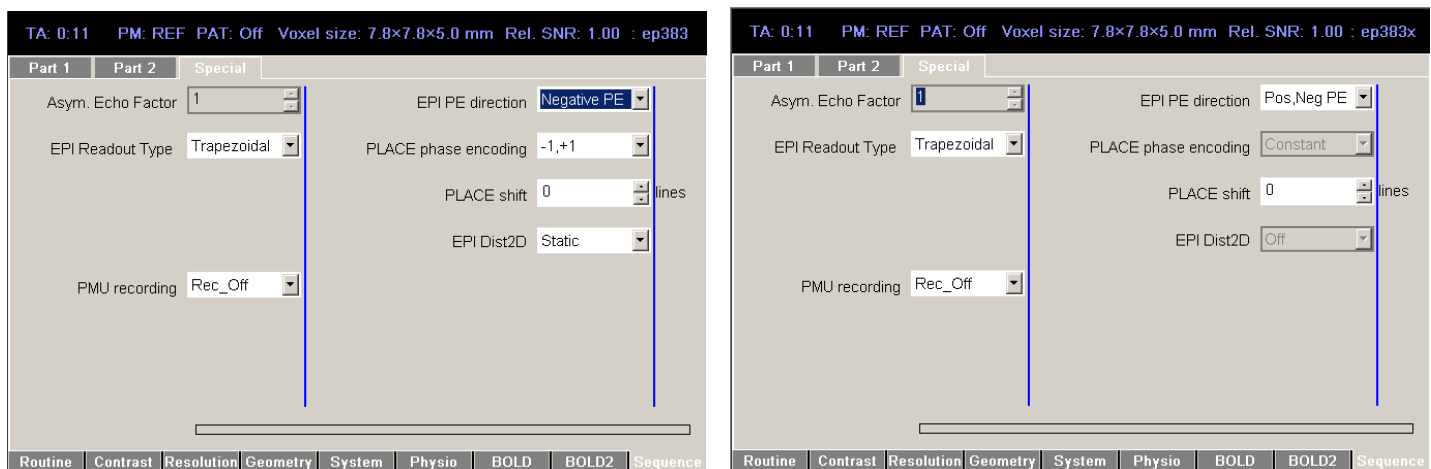
Figure: Sequence Tab with selected phase encoding direction (neg) using PLACE algorithm (left); alternating phase encoding direction using Gradient Reversal algorithm (right)

XXXXXXXXXXXXXXXXXXXXXX

You must select **Magnitude/Phase** mode on the Contrast tab.

GLM statistics and Motion correction is only possible **offline**.

Known bug: Do not use the APPLY button to start scanning, use SCAN

XXXXXXXXXXXXXXXXXXXXXX

The distortion correction is divided into three major parts.

1. Data acquisition with a modified EPI sequence.

2. Calculation of the displacement map which contains the voxel shift

3. Application of the displacement map to shift the voxels to their original positions

Configuration of EPI Distortion Correction

Within the first execution of the Ice-Program a new file C:\MedCom\config**EPIDistorCorConfig.evp** will be created.

It contains some parameters to control the correction method.

List of Parameters in EPIDistorCorConfig.evp

- **bSendDmap**
  - send the calculated displacement to the host
  - default **true**

- **bApplyDmap**
  - if this value is "true" an additional series with corrected images will be created
  - default **true**

- **lInterpolationMode**
  - specifies the interpolation algorithm to realize subvoxel shifts
  - **1 = Linear, 2 = Spline**
  - default: 1

- **bApply2DSmoothing**
  - if this value is "true" an additional 2d smoothing filter is applied to reduce artifacts
  - default **false**

- **lExpansion**
  - specifies the ratio by which the images are expanded in phase-encoding direction (for subvoxel shifts)
  - default **100**

- **lIsocenterOffset**
    - Specifies an additional offcenter shift in phase encoding direction., which is needed by the PLACE algorithm. Normally the shift is calculated correctly by the framework. **Only change this parameter if you know what you are doing.**
    - default **0.0**
- **dColorLookup**
    - Specifies a ratio to scale the displacement map at the Syngo host. Normally values between 0..4095 are allowed in Syngo. This means that the displacement map must be scaled to these values.
    - 2048 = no shift
    - 0..2047 = negative shift
    - 2049..4095 0 positive shift
    - lIsocenterOffset is a divisor to increase this range
    - example1: lIsocenterOffset = 10
        - 2049 represents a shift of +1/10 voxel
    - example2: lIsocenterOffset = 20
        - 3048 represents a shift of
        - (3048-2048)/20 = 50 voxel
    - default **10**
- **bWrite2dseq**
    - if this value is "true" the raw complex data is written to disk
    - real part filename: **2dseq**
    - imaginary part filename: **2dseq.im**
    - the data is written binary float in column, row, slice, repetition order
    - default **false**
- **bWriteDmap**
    - if this value is "true" the displacement maps are written to disk **for debugigng**
    - for every slice a new displacement map is written
    - example: slice=2 repetition=5
        - filename = dmapr5s2

- o default **false**

- **bCalcStatistics**
  - o If this value is "true" the first 5 repetitions are used to compute statistics about the stability of the input images. Then, the "best" pair of images is selected to compute the displacement map.
  - o min 6 repetitions are nedded
  - o works only with PLACE -1, +1
  - o default **false**

- **lMaxThreshold**
  - o If PLACE is selected, an automatic image threshold is performed. This value defines the maximum threshold. Values from 0..4095 are possible
  - o default **100**

- **lFixedThreshold**
  - o If this value is not zero, the automatic image threshold is overwritten by this value. Values from 0..4095 are possible.
  - o default **0**

- **dLogisticalSlope**
  - o If PLACE is selected, an automatic image threshold is performed. The threshold function uses a logistic curve to separate the object from noise. dLogisticalSlope specifies the slope of this logistic function.
  - o default **0.01**
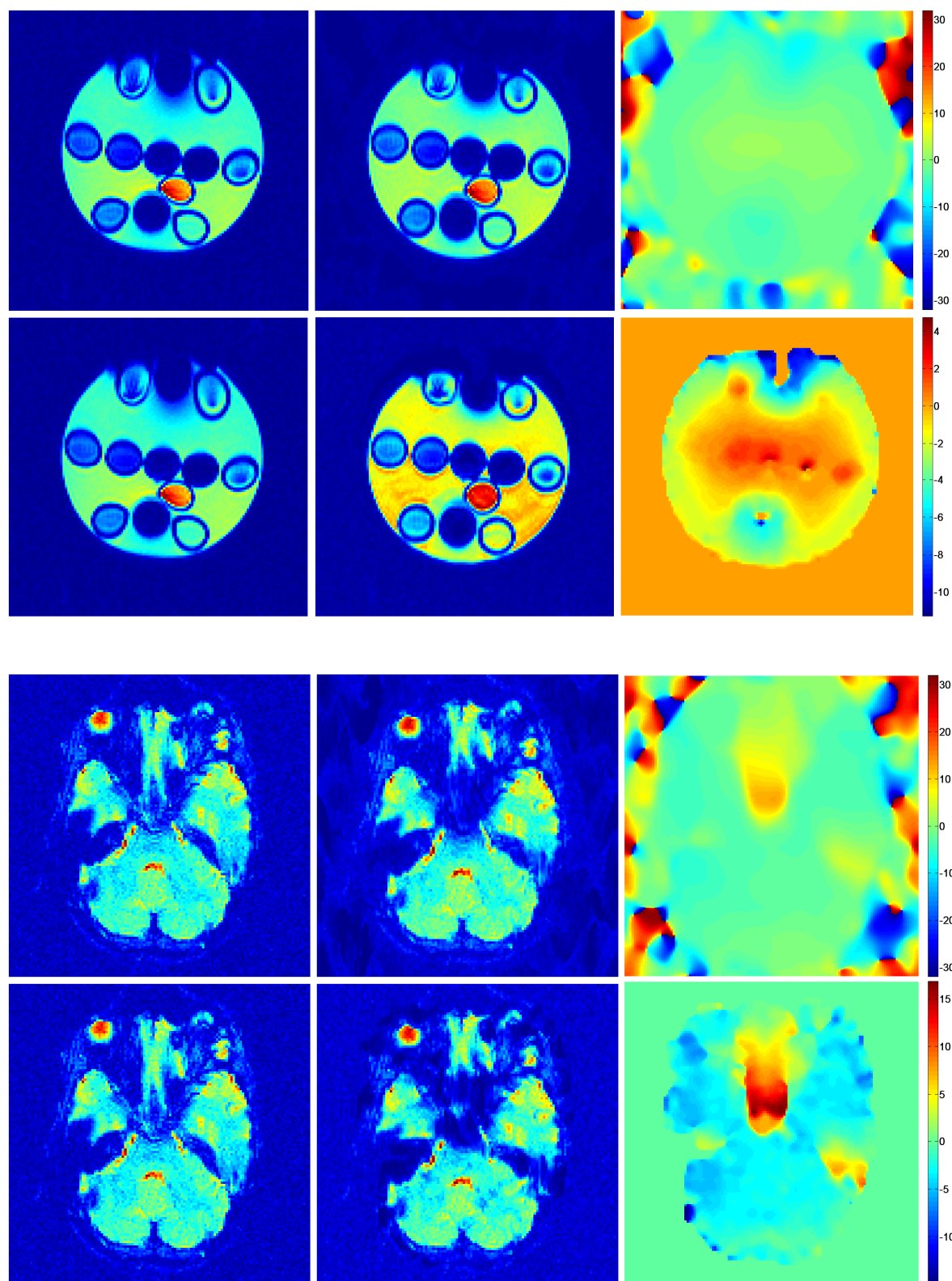
Figure: **Results of modified PLACE algorithm (phantom / human):**
Original PLACE algorithm (top), modified algorithm (bottom);
Input (left), corrected image (middle), displacement map (right)
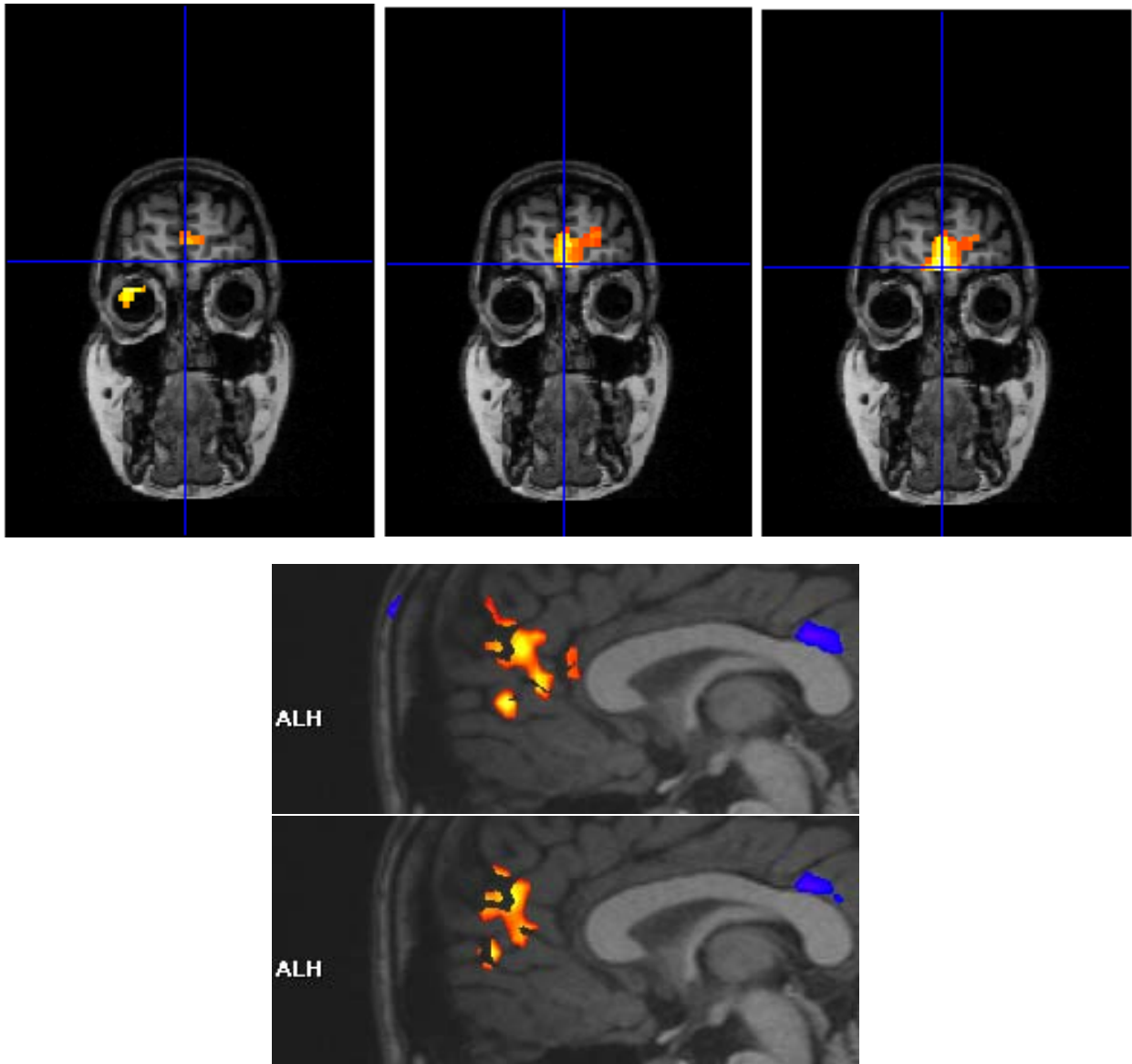
Figure: **Distortion Correction in fMRI Experiments:**
Before correction (left), PLACE correction (middle), fieldmap based correction (right)
from: Inline Distortion Correction for Echo-Planar fMRI, . Vogler, S. Arnold, O. Hinds, S. Whitfield-Gabrieli, J. Pfeuffer, C. Triantafyllou, Hum. Brain Mapping 2008

5. Physio Logging

Description of Physio Log Files

The following signals can be recorded from the BOLD sequence: ECG, Respiration, and Pulse.   The log files will be available in C:\Medcom\log.   Three different files are generated if the Physio Signal logging checkbox is selected in the sequence.   Each filename includes the timestamp of when it was generated, therefore will be unique.   No files are overwritten.   The users are encouraged to copy the files from C:\Medcom\log and then delete them from the above location.

The format of the data stream in the log files is given below:

The first 4 values are don't care.

- addtional header data between 5002 and 6002 are also don't care

- The maximum value of the data is 4095, the minimum is 0, 2048 is the zeroline -> 12 bit resolution

- Software versions VAxx: 1 ECG channel, range as above indicated

- Software versions VB1x: 2 ECG channels, interleaved storage in the file:

Channel I: The maximum value of the data is 4095, the minimum is 0, 2048 is the zero line.

Channel aVF: The maximum value of the data is 12287, the minimum is 8192, 10240 is the zero line.

- Sample frequencies: ECG 400 Hz, Pulse 50 Hz, RESP 50 Hz.

- 5000 = Trigger on (rising edge, relevant), 6000 = Trigger off (just for visualization purposes) – filter both these out when using data for post-processing.

- 5003 = end of data, followed by the statistics data at the end of the file.

short term statistics (current short term averages at the end of the data logging):

ECG Freq Per: 0 0

PULS Freq Per: 0 0

RESP Freq Per: 0 0

EXT Freq Per: 0 0

long term statistics (long term averages etc. from the begin of the last protocol till the end of the data logging):

ECG Min Max Avg StdDiff: 0 0 0 0

PULS Min Max Avg StdDiff: 610 2503 959 19

RESP Min Max Avg StdDiff: 0 0 0 0

EXT Min Max Avg StdDiff: 0 0 0 0

NrTrig NrMP NrArr AcqWin: 49 0 0 92

Synchronization

There is a small time difference between signal logging and the scan at the beginning of a scan, and a larger time delay after the end of the scan. The signal logging starts before the scan does and ends after the scan ends. The best method to estimate the delay at the beginning of the scan is to plot the ECG signal, and look for gradient disturbances when the sequence starts. Counting the number of samples (=duration of the scan x sampling frequency) will define the end point of scan in the physio signal log file.

6. General Linear Model Statistics.

During an fMRI measurement a large number (roughly 10^2-10^4) of brain volume-datasets are acquired using the appropriate sequences and contrast-parameters. As a result of the BOLD contrast mechanism during changes in the local neuronal activity, signal fluctuations in the order of a few percent can be detected in the image data.

When evaluating the fMRI data, the main question to answer is, whether there is a statistically relevant relationship between an assumed time-course model and the measured time series during local neuronal activity. In the simplest case, this model (or reference function) is identical to the stimulation scheme or paradigm; therefore the transition states between the single states are inadequately defined. However, assuming that the brain behaves like a linear system, by using a hemodynamic response function (HRF), a better model of the time courses in case of activity can be obtained. Mathematically this requires a convolution of the paradigm with the HRF. A block-design paradigm (i.e. 30 sec. Stimulus – 30 sec pause …) can be entered by the operator using the BOLD-tabcard of the exam-UI when setting up the measurement protocol for a BOLD-sequence.

In contrast to a simple t-test or correlation, the GLM framework allows a multiple number of reference functions to be entered. These reference functions may describe certain aspects of the data such as the hemodynamic response of the brain to a stimulus. In addition to these covariates of interest it is possible to add more information into the model; for example knowledge about physiological or system related signal fluctuations, also known as covariates of no interest (which means they are present to describe the data, though we are not interested in their behaviour). By using multiple model functions, a better description of the measured voxel time-courses is possible enabling a better detection power of the statistics. Prerequisite for a reasonable design is a sufficient orthogonality of the model time courses (i.e. the variance/covariance matrix should be diagonal).

All desired model time-courses are put as columns into a so called design matrix G. Using a contrast vector C, the interesting components can be marked (i.e. 1/0/0/0...) or contrasted against each other (i.e. 1/-1/0/0...).

As result of the GLM analysis a t-map is generated, similar to the current Student's t-Test statistics.

Integration into the product framework

The calculation of the GLM is performed at the end of the ICE postprocessing chain within an EVA-Functor (EVA means "evaluation" and can be run online during the scanning or offline). The mechanism of the GLM calculation is identical to the existing concept of calculating a t-test, while the algorithm was replaced and new parameters were introduced. The new algorithm continues to generate t-maps and superimposed data in a similar way as before, so that the new functionality can be integrated seamlessly including all options like inline, post-processing and visualization. The current product functionality is briefly described below:

Starting ignore:	Number of Measurements **at start** which are excluded from the GLM statistics
Ignore after trans:	Number of Measurements **after each change** of paradigm state (Baseline $\rightarrow$ Active, Active $\rightarrow$ Baseline) which are excluded
Model transition:	Convolute paradigm with a hemodynamic response function
Temp highpass:	Add covariates of no interest, which model temporal baseline
Threshold:	for the inline display of the t-map

Paradigm size: Length of the paradigm, is repeated by circular rotation for the whole number of Measurements

Paradigm: Defines the functional states: Baseline / Active
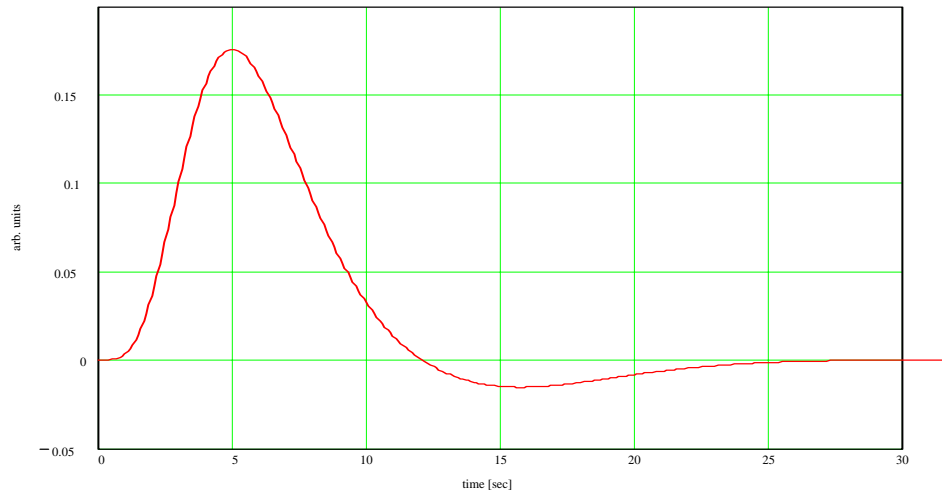
Typical characteristics of a hemodynamic response.

Single-condition *paradigms* can be currently set by the user-interface with the states "active" and "baseline". This concept is maintained, while the currently used possibility to enter the state "ignore" will be removed. The "ignore" have been used previously in the classical t-test option to remove transition volumes between "active" and "baseline" states from the data analysis. However, within the GLM framework, we have the possibility to model these transition states by the use of a *hemodynamic response function (*HRF), and therefore the concept of an "ignore" state in the paradigm is not useful any more. Furthermore, to perform convolution of the paradigm with the HRF, for every time-point, the stimulation condition must be known. To maintain the possibility to remove transition states from the analysis, a new parameter "*Ignore after Transition*" is offered. In addition to the above mentioned features, the user also can enable a "*temporal high-pass filter*" operation. This filter models low frequency drifts present on the data. An analysis of the paradigm for the longest cycle length allows the automatic selection of an appropriate cut-off frequency.

Construction of the design-matrix

Within the GLM framework, a better description of the measured data could be obtained using several model functions. The values of the model functions are added as columns into a design matrix G.

Combining all components of the design matrix results in the following representation:

Covariates
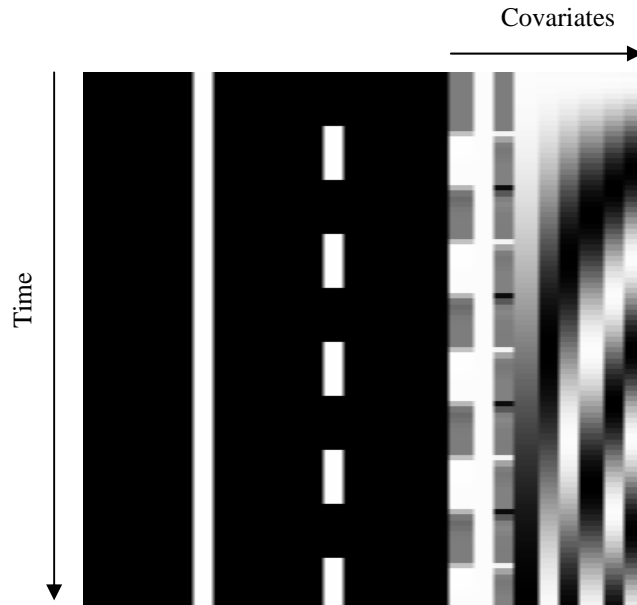
Typical design using 100 measurements:
left: used (white) and unused (black) data   (all are used here)..
middle: paradigm describing the stimulus states
right: design matrix consisting of (left to right) convoluted paradigm,
offset, derivative of stimulation model, 6 components highpass filter

A similar image is produced and saved to the DICOM database at the end of a GLM evaluation to provide a control of the performed statistics to the user.

A trivial component of the design matrix is a column that contains a constant value. This describes the offset of the voxel time-course, which by default does not have zero mean. One column describes the expected BOLD response to the paradigm. Without using the option to model the reference data using the HRF, simply a copy of the paradigm is used here. In case of modelling by HRF, a convolution with an identical HRF as used in SPM2 (http://www.fil.ion.ucl.ac.uk/spm) or FSL (http://www.fmrib.ox.ac.uk/fsl) is performed.

The HRF consists of two parts: Firstly a delayed, positive overshoot, followed by an even more delayed undershoot. A graphical representation of the HRF used in this package is shown below:

The acquisition of a volume dataset (i.e. typically a 2D multi-slice experiment) takes about 2-5 seconds. During this period the single slices are acquired at different time-points. To confound for a temporal shift of the measured data compared to the model, a further column is added to the design-matrix. A *shifted model-function* can be obtained in a good approximation by adding its first temporal derivative to it.

In addition to this option, the user can select a *temporal high-pass filter* for the GLM evaluation. In this case low frequency drifts are included in additional columns as covariates of no interest. It is important to take care that the highest frequency component of the filter is set sufficiently below the lowest frequency components of the paradigm, to assure an adequate orthogonality of the columns of the design matrix. This is handled automatically by analyzing the paradigm.

Calculation of GLM results

Using a design matrix, like the one presented in the previous paragraph, one chooses the contrast vector c=(1,0,0,0,0,0,0,0,0), i.e. only the first column in the design matrix, to consider only the model of the neuronal stimulation ("baseline" vs. "active") as interesting.

The first step in the GLM calculation is the estimation of the linear combination of the columns of the design matrix G, which describe the data best. The factors creating this linear combination are put into a vector b.

If we would have this vector b already, we could describe the measured time domain voxel vector x using the following relationship:

$$G \cdot \vec{b} = \vec{x} + \overrightarrow{Error}$$

Using a standard mathematical procedure a least squares solution for b is calculated. To calculate a t-value, the noise relevant for the covariates of interest must be considered against the signal changes observed in vector b:

$$t = \frac{\vec{c} \cdot \vec{b}}{\sqrt{\sigma^2 \cdot \vec{c} \cdot \left(G^T \cdot G\right)^{-1} \cdot \vec{c}}}$$

For a more complete description of the math behind the GLM you may consider the following literature:

R.S.Frackowiak et al.; Human Brain Function ; Academic Press 1997

P. Jezzard et al.; Functional MRI – an introduction to methods ; Oxford Univ. Press 2001

To calculate a GLM statistics inline during the acquisition, the algorithm in this package has been reformulated to an incremental algorithm. For generating a t-map for every voxel the time course is analyzed as described above.

Why use GLM ?

There are a number of reasons for using the GLM approach to evaluate fMRI data:

- Data acquired in transition states is not lost for the analysis.
- Temporal shifts can be modelled.
- Low frequency drifts can be removed from the data providing a suppression for motion as well as signal drifts, which can e.g. be caused physiological effects.

A comparison between the GLM and t-statistics is illustrated in the figure below. The generated t-maps calculated from an alternating left/right-handed finger tapping fMRI experiment with relatively low head motion. Motion correction has been used before evaluating the data. Clearly the GLM-based analysis performs better and provides less background noise.
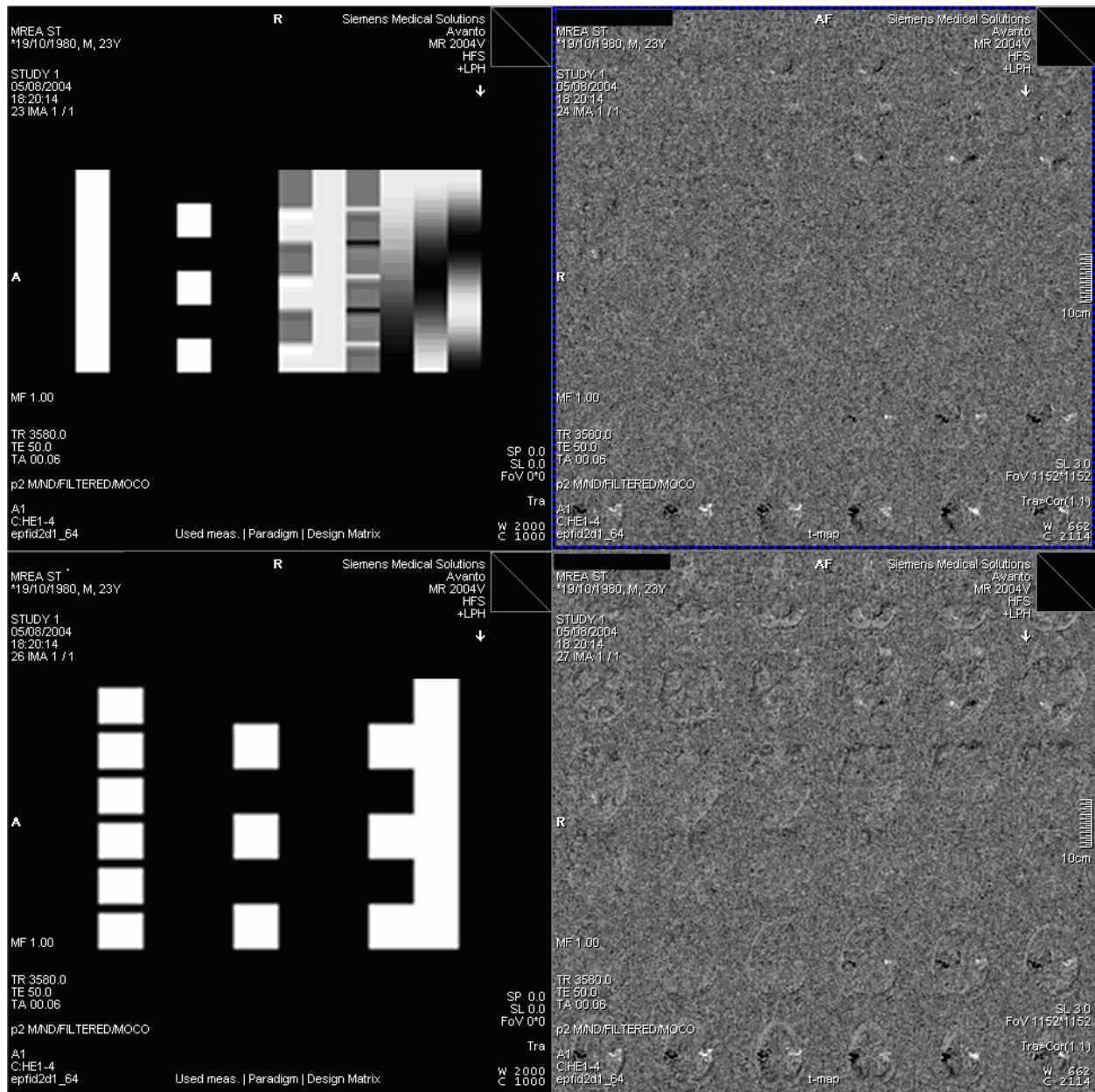
Comparison of a GLM-analysis (top row) and a t-test (bottom row; result identical to standard t-test result). The GLM analysis picks up the active areals more efficiently and shows less sensitivity to residual head motion effects (borders of brain less visible).

BOLD2 card: Inline User Interface for extended GLM features

To enable the extended GLM features, the user-interface is extended for the inline-case and has additional parameters on the new tabcard 'BOLD2' on the exam UI:

The so-called "External Design" is activated by the checkbox "UseExternalDesign", which sets the control by an GLM-ini file "ExtDesignName" (see below).

*** **BE AWARE** *** that the settings in the GLM-ini file can overwrite and redefine nearly ALL the parameter settings on the standard 'BOLD' tabcard, such as:

> *Starting ignore, Ignore after trans, Model transition, Temp highpass, Threshold, Paradigm size*, and *Paradigm.*

Therefore it is inevitable in practice that the corresponding GLM-ini file is linked and stored for a specific fMRI data set in order to possible reproduce the GLM analysis in offline processing. The most important parameters used the GLM statistics can be checked in the created Design-image in the DICOM base, which is generated at the end of every evaluation containing the design matrix.

Offline User Interface

The same analysis that can be performed inline is also available as offline processing by selecting 'BOLD Evaluation' from the patient browser. Here the configuration parameters are not entered by a graphical user interface but by editing evaluation protocols.

The same options with the same names as described previously are available in the eva-protocols that begin with 'glm'. Several template GLM evaluation protocols are delivered with this package and can be used as a template.

Using external design files (GLM-ini files) for advanced GLM statistics

The graphical user interface (BOLD / BOLD2 card) and the evaluation protocols (glm*.evp) allow to set-up a customized GLM evaluation for one or more contrasts, multiple conditions or factorial designs etc using an external design file (GLM-ini file).

```
┌─────────────────────┐      ┌──────────────┐      ┌──────────────────────┐
│    BOLD card:       │─────▶│     GLM      │◀─────│    BOLD2 card:       │
│  single condition   │      │ Calculation  │      │  ExtDesignName.ini   │
└─────────────────────┘      └──────────────┘      └──────────────────────┘
```
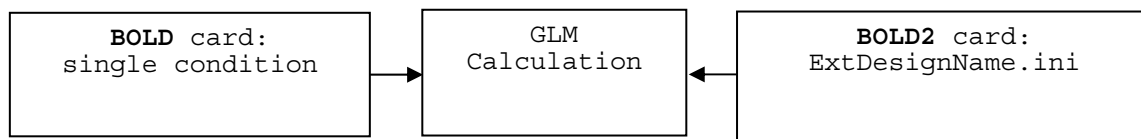
The so-called "External Design" (in contrast to the built-in design on the BOLD card with a single condition and a single contrast) is activated by the checkbox "UseExternalDesign" and the setting of an GLM-ini file "ExtDesignName", which located on the host computer at

```
C:\MedCom\MriCustomer\GLM_Designs\*.ini
```

GLM-ini file format

An INI file contains ASCII configuration data for the GLM analysis. A typical INI file might look like this (see e.g. http://en.wikipedia.org/wiki/Initialization_file):

```
[section1]
# some comment on section1
var1 = foo
var2 = doodle

[section2]
# another comment
IgnoreIndicesLines    = 2
IgnoreIndicesColumns  = 4  # maximum is 10
IArr[1]=  0 0 0 0
IArr[2]=  1 2 3 4
```

and contains **Section** declarations (start with '[' and end with ']'),

Parameters (begin with a key ('var1'), equals sign ('='), and a value ('foo'),

Comments (starting with a ';' or '#') which are ignored, and

2D Matrices (with a key ('IArr'), index ('[num]'), an equals sign ('='), values (' 1 2 3 4').

The dimensions are declared by a xLines & xColumns parameter:

```
Dmat[]:    DesignMatrixLines, DesignMatrixColumns

DTMat[]:   DesignTMatrixLines, DesignTMatrixColumns

Cmat[]:    ContrastMatrixLines, ContrastMatrixColumns

CDMat[]:   ConvDerivMatrixLines, ConvDerivMatrixColumns

Iarr[]:    IgnoreIndicesLines, IgnoreIndicesColumns
```

Note: The maximum number of columns in a row of the 2D matrix is 10 !

GLM-ini file description

First some explanations of the terminology for the description of a paradigm: An fMRI experiment can be run with a single stimulus **condition** ("Active" vs. "Baseline") or multiple stimulus conditions (cond1: "visual:Active" vs. "Baseline", cond2: "motor:Active" vs. "Baseline", … cond X: ...). For an experiment with multiple conditions, different statistical t-maps can be calculated, which are referred to as different **contrasts** as defined in the contrast matrix. A contrast is typically a linear combination (factors -1, 0, 1) of the different stimulus conditions.

The GLM-ini file is structured in the following sections:

[Design]

Variables can be set here: t-threshold, StartInlineAtMeas, StartingIgnores, HighpassCutOffCycle, HRFDelay_s (all optional)

[DesignMatrix]

Matrix to define the covariates used in the GLM analysis. The amplitude of the conditions can be float values, e.g. 0, 0.25, 0.5 0.75, 1.

Dimensions (row x col):   time (volume-based) x number of conditions

[DesignTMatrix] (optionally)

Variables: DesignTimeBase (in [s], if zero: volume-based), DesignTimeShift (in [s], optional)

Dimensions: time x (time onset, time duration, number of conditions)

[ContrastMatrix]

Matrix to define the different contrasts for combing different conditions

Dimensions: number of contrasts x number of conditions

[ConvDerivMatrix]

Matrix to set flags for HRF convolution (row1) and adding derivatives (row2)

Dimensions: 2 x number of conditions

[IgnoreIndices]

Array to define indices of volumes in the DesignMatrix that are ignored for GLM calculation in the time series. Indices are running from   [1...Timepoints] in the DesignMatrix.

All files delivered as a template with this package contain comments on how to use and configure an experiment. It is clearly desired, that you create your own experimental designs starting from the example design folders. All files can be edited using a text editor like 'wordpad' available on the user console.

To enable the GLM statistics via the external input file, activate the parameter '*Use External Design*' on the **BOLD2 card**. A new parameter '*ExtDesignName''* will appear: Enter the name of GLM-ini file. Note that only a limited amount of characters that can be entered in the inline graphical user interface. The name of the sub-directory must fit in here.

In case of **offline** processing (i.e. for BOLD Evaluation) the EVP file must be edited manually: set the parameter '*UseExternalDesign*' to '*true*' and specify the GLM-ini file name in the section of parameter '*ExternalInputDir*.

All kinds of contrasts can be specified to look at differential effects as well as summary effects. If you are looking for an F-test, please consider the following relationship between F- and t-values:

$$F = \frac{t^2}{k}$$

where k is the number of covariates you are testing for. This package always outputs t-maps to allow for a consistent visualization.

If you have syntax errors in the GLM-ini file, the scan or offline evaluation will not start. In that case:
- Check the name of the GLM-ini file and its correct location.
- Carefully check for typos in the INI file.
- Check that the number of covariates is identical in all sections.
- You can find concrete hints in the output of log-files

        Run: logviewer C:\MedCom\log\MrParcTraceSrv.log
    Note: this is probably only useful for advanced users.

Example: Paradigm with single stimulus condition, single contrast

This paradigm can be e.g. a visual task (cond1: flicker vs. blank) or a motor task (cond1: finger movement vs. rest), resulting in a t-map (single contrast):

Condition 1

10 vols   10 vols
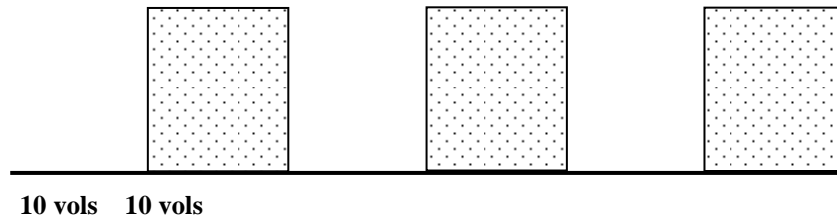
In the example file <p1cond.ini>, the design matrix is entered in the **section [DesignMatrix]** with image volume-based timing (sequence repetition time TR), whereas the stimulus level is set to 0.0 (baseline) and 1.0 (active) in the first column (condition 1). Always add a constant offset (col 2) to improve the model. If more measurements occur than lines specified (which is usually the case), the matrix is used in a cyclic way.

In the example file <p1condDT.ini> the design matrix is alternatively entered in the **section [DesignTMatrix]** with image volume-based timing (set *DesignTimeBase* = 0). The matrix format contains now two additional columns: *time onset* (col 1) and *time duration* (col 2). The 3rd and 4th column contain condition 1 and a constant offset.

The **section [ContrastMatrix]** defines how many contrast (given in the rows) are calculated, which are linear combination (factors -1, 0, 1) of the different stimulus conditions. For a single condition paradigm, typically only a single contrast is calculated. For a dual condition paradigm, up to four contrasts can be calculated. The first contrast (first row) is used in the real-time calculation for Neuro3D 'Inline mode'.

The **section [ConvDerivMatrix]** defines, whether the corresponding condition is convolved with the HRF model function (row 1) and whether a first temporal derivative of the condition is added to the DesignMatrix to compensate for time jitter between stimulus and hemodynamic response (row 2).

Example: Paradigm with two stimulus conditions, four contrasts

This paradigm can be e.g. a combined visual-motor task (cond1: flicker vs. blank, cond2: finger movement vs. rest) or a dual motor task (cond1: left hand vs. rest, cond2: right hand vs. rest).

Condition 1

10 - 10 - 10 vols

Condition 2

10 - 10 - 10 vols
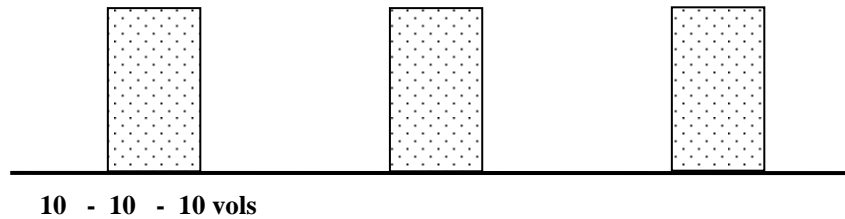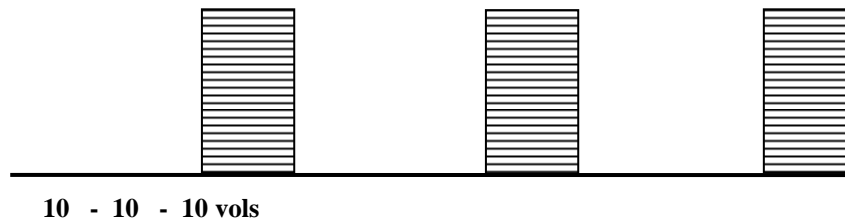
From two conditions, four different contrasts (t-maps) can be calculated as defined in the the (**section [ContrastMatrix]**:

	Contrast 1	Contrast 2	Contrast 3	Contrast 4
Condition 1	+1	+1	+1	0
Condition 2	+1	−1	0	+1

C1 is the sum of both conditions, C2 is cond1 contrasted against cond2 (positive t-values for cond1 > cond2, negative t-values vice versa), C3 is cond1 only, and C4 is cond2 only. The first contrast C1 is used in the real-time calculation for Neuro3D 'Inline mode'.

In example file <p2condDT.ini>, the design matrix is entered in **section [DesignTMatrix]** with image volume-based timing (set *DesignTimeBase* = 0). The five columns contain time onset, time duration, condition1, condition2, and a constant offset.

Example: Event-related paradigm with arbitrary stimulus time base

This paradigm serves as an example for an event-related paradigm with stimuli of different amplitude and arbitrary time-base (uncoupled from the EPI volume TR).

Condition 1 with 2 short stimuli — 100% — 50% — 200 100 ms
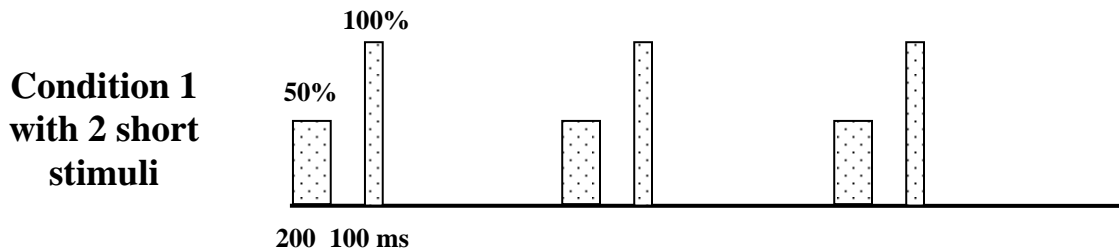
In the example file <p1short.ini>, the design matrix is entered in the **section [DesignTMatrix]** with a time base of 100 ms (*DesignTimeBase* = 0.1 [s]). The onset of the events are at t = 0 s and 10 s, the end of the epoch is at t = 40 s (col 1). The stimulus duration is 200 ms and 100 ms, respectively (col 2). The stimulus level is 50 and 100, respectively, representing for example a visual flicker stimulus with 50% and 100% contrast (col 3).

In the time-base mode (*DesignTimeBase* > 0) several conditions have to be fulfilled:

1)   The image TR must be a multiple of *DesignTimeBase* (e.g. DesignTimeBase = 0.5 [s], TR = 2100 ms is not valid: change DesignTimeBase to 0.1 [s])

2)   Time onset and time duration (col 1, col2 in DesignTMatrix) must be a multiple of *DesignTimeBase* (e.g. Onset of 0.125, DesignTimeBase = 0.1 are not compatible)

3)   The time onset in a line of DesignTMatrix (col 1) must be larger than the sum of time onset and duration (col 1 + col 2)

After the construction of the stimuli on the defined time base, HRF convolution is first applied and subsequently resampled to the final DesignMatrix on the volume-based time line (given by the image TR).

```
#-------------------------------------------------------------------------------
# GLM Paradigm Input File <p1cond.ini> Josef Pfeuffer Aug-2006
#-------------------------------------------------------------------------------
[Design]
t-threshold        = 4          # [typical: 4]  [range: >=0]  [optional]
StartInlineAtMeas  = 20         # [typical: 20] [range: >0]   [optional]
StartingIgnores    = 0          # [typical: 0]  [range: >=0]  [optional]
#HighpassCutOffCycle = 25       # [typical: 20] [range: >0]   [optional]
                                # switch off highpass-filter with 99999.
#HRFDelay_s         = 0         # [typical: TR/2] [range: +/-TR] [optional]


#-------------------------------------------------------------------------------
[DesignMatrix]
#         Note: always ADD A CONSTANT COLUMN (ie. offset) to model the data correctly.
#         If more measurements occur than lines specified (which is usually the case),
#         the matrix will be used in a cyclic way.
DesignMatrixLines      = 20
DesignMatrixColumns    = 2      # MUST be identical to the other two matrices !!!

#         Cond1 Offset
DMat[1]=  0     1               # Measurements [1..10] are Cond1: baseline
DMat[2]=  0     1
DMat[3]=  0     1
DMat[4]=  0     1
DMat[5]=  0     1
DMat[6]=  0     1
DMat[7]=  0     1
DMat[8]=  0     1
DMat[9]=  0     1
DMat[10]= 0     1
DMat[11]= 1     1               # Measurements [11..20] are Cond1: active
DMat[12]= 1     1
DMat[13]= 1     1
DMat[14]= 1     1
DMat[15]= 1     1
DMat[16]= 1     1
DMat[17]= 1     1
DMat[18]= 1     1
DMat[19]= 1     1
DMat[20]= 1     1

#-------------------------------------------------------------------------------
[ContrastMatrix]
#         Note: if you are contrasting conditions against each other,
#         the sum of the contrast-vector should be zero.
ContrastMatrixLines   = 1
ContrastMatrixColumns = 2       # MUST be identical to DesignMatrixColumns

#         Cond1 Offset
CMat[1]=  1     0               #

#-------------------------------------------------------------------------------
[ConvDerivMatrix]
#         Note: Careful with the derivatives! If you derive a constant column or a box-car
#         shape, the result will not be useful.  Values 0 and 1 only!
#         The last column value should usually be zero (ie. constant offset).
ConvDerivMatrixLines   = 2
ConvDerivMatrixColumns = 2      # MUST be identical to DesignMatrixColumns

#         Cond1 Offset
CDMat[1]= 1     0               # switch for HRF convolution (0/1 only)
CDMat[2]= 1     0               # switch to add derivative  (0/1 only)

#-------------------------------------------------------------------------------
[IgnoreIndices]
#         Define indices of volumes that are ignored for GLM calculation in the time series.
#         Indices are running from  [1...Timepoints].
#         Indices >Timepoints or <1 are NOT considered for <IArr>.
IgnoreIndicesLines    = 2       #
IgnoreIndicesColumns  = 4       # maximum is 10

IArr[1]=  0 0 0 0
IArr[2]=  0 0 0 0
#------------------------------------------------------------------------- end
```

```
#-------------------------------------------------------------------------------
# GLM Paradigm Input File <p1condDT.ini> Josef Pfeuffer Aug-2006
#-------------------------------------------------------------------------------
[Design]
t-threshold         = 4              # [typical: 4]  [range: >=0]  [optional]
StartInlineAtMeas   = 20             # [typical: 20] [range: >0]   [optional]
StartingIgnores     = 0              # [typical: 0]  [range: >=0]  [optional]
#HighpassCutOffCycle = 25            # [typical: 20] [range: >0]   [optional]
                                     # switch off highpass-filter with 99999.
#HRFDelay_s          = 0             # [typical: TR/2] [range: +/-TR] [optional]


#-------------------------------------------------------------------------------
[DesignMatrix]
#        Note: always ADD A CONSTANT COLUMN (ie. offset) to model the data correctly.
#        If more measurements occur than lines specified (which is usually the case),
#        the matrix will be used in a cyclic way.

DesignMatrixLines    = 1
DesignMatrixColumns  = 2           # MUST be identical to the other two matrices !!!

#        Cond1 Offset
DMat[1]= 0     1                   # dummy line: see DesignTMatrix

#-------------------------------------------------------------------------------
[DesignTMatrix]   # [optional] if defined: replaces [DesignMatrix]
#        Note: amplitudes are preset to 0, last covariate is preset to 1 (offset);
#        time base can be: volumeBase (TR)  or timeBase [s]!
#        stimuli are convoluted with HRF and resampled to volume TR for GLM

DesignTimeBase        = 0          # [s] – set 0 for volumeBase (TR)
#DesignTimeShift      = 0          # [s] [optional]
DesignTMatrixLines    = 2
DesignTMatrixColumns  = 4          # MUST be (DesignMatrixColumns + 2)

#----------- The DesignTimeMatrix defines time points, time (first volume) starts at ZERO !
#        Onset Dur   Cond1 Offset
DTMat[1]= 0    10    0     1       #
DTMat[2]= 10   10    1     1       # cond1:active

#-------------------------------------------------------------------------------
[ContrastMatrix]
#        Note: if you are contrasting conditions against each other,
#        the sum of the contrast-vector should be zero.
ContrastMatrixLines    = 1
ContrastMatrixColumns  = 2         # MUST be identical to DesignMatrixColumns

#        Cond1 Offset
CMat[1]= 1     0                   #

#-------------------------------------------------------------------------------
[ConvDerivMatrix]
#        Note: Careful with the derivatives! If you derive a constant column or a box-car
#        shape, the result will not be useful.  Values 0 and 1 only!
#        The last column value should usually be zero (ie. constant offset).
ConvDerivMatrixLines    = 2
ConvDerivMatrixColumns  = 2        # MUST be identical to DesignMatrixColumns

#        Cond1 Offset
CDMat[1]= 1     0                  # switch for HRF convolution (0/1 only)
CDMat[2]= 1     0                  # switch to add derivative  (0/1 only)

#-------------------------------------------------------------------------------
[IgnoreIndices]
#        Define indices of volumes that are ignored for GLM calculation in the time series.
#        Indices are running from  [1...Timepoints].
#        Indices >Timepoints or <1 are NOT considered for <IArr>.
IgnoreIndicesLines    = 2          #
IgnoreIndicesColumns  = 4          # maximum is 10

IArr[1]=  0 0 0 0
IArr[2]=  0 0 0 0
#----------------------------------------------------------------------------- end
```

```
#--------------------------------------------------------------------------------
# GLM Paradigm Input File <p2condDT.ini> Josef Pfeuffer Aug-2006
#--------------------------------------------------------------------------------
[Design]
t-threshold           = 4              # [typical: 4]  [range: >=0]  [optional]
StartInlineAtMeas     = 20             # [typical: 20] [range: >0]   [optional]
StartingIgnores       = 0              # [typical: 0]  [range: >=0]  [optional]
#HighpassCutOffCycle  = 25             # [typical: 20] [range: >0]   [optional]
                                       # switch off highpass-filter with 99999.
#HRFDelay_s           = 0              # [typical: TR/2] [range: +/-TR] [optional]


#--------------------------------------------------------------------------------
[DesignMatrix]
#        Note: always ADD A CONSTANT COLUMN (ie. offset) to model the data correctly.
#        If more measurements occur than lines specified (which is usually the case),
#        the matrix will be used in a cyclic way.
DesignMatrixLines     = 1
DesignMatrixColumns   = 3              # MUST be identical to the other two matrices !!!

#        Cond1 Cond2 Offset
DMat[1]= 0     0     1                 # dummy line: see DesignTMatrix


#--------------------------------------------------------------------------------
[DesignTMatrix]   # [optional] if defined: replaces [DesignMatrix]
#        Note: amplitudes are preset to 0, last covariate is preset to 1 (offset);
#        time base can be: volumeBase (TR)  or timeBase [s]!
#        stimuli are convoluted with HRF and resampled to volume TR for GLM
DesignTimeBase        = 0              # [s] - set 0 for volumeBase (TR)
#DesignTimeShift      = 0              # [s] [optional]
DesignTMatrixLines    = 3
DesignTMatrixColumns  = 5              # MUST be (DesignMatrixColumns + 2)
#---------- The DesignTimeMatrix defines time points, time (first volume) starts at ZERO !
#        Onset Dur   Cond1 Cond2 Offset
DTMat[1]= 0     10    0     0     1
DTMat[2]= 10    10    1     0     1     # cond1:active
DTMat[3]= 20    10    0     1     1     # cond2:active


#--------------------------------------------------------------------------------
[ContrastMatrix]
#        Note: if you are contrasting conditions against each other,
#        the sum of the contrast-vector should be zero.
ContrastMatrixLines   = 4
ContrastMatrixColumns = 3              # MUST be identical to DesignMatrixColumns


#        Cond1 Cond2 Offset
CMat[1]= 1     1     0                 # sum of cond1 and cond2
CMat[2]= 1     -1    0                 # cond1 and cond2 are contrasted against each other
CMat[3]= 1     0     0                 # cond1 only
CMat[4]= 0     1     0                 # cond2 only


#--------------------------------------------------------------------------------
[ConvDerivMatrix]
#        Note: Careful with the derivatives! If you derive a constant column or a box-car
#        shape, the result will not be useful.  Values 0 and 1 only!
#        The last column value should usually be zero (ie. constant offset).
ConvDerivMatrixLines   = 2
ConvDerivMatrixColumns = 3             # MUST be identical to DesignMatrixColumns

#        Cond1 Cond2 Offset
CDMat[1]= 1    1     0                 # switch for HRF convolution (0/1 only)
CDMat[2]= 1    1     0                 # switch to add derivative  (0/1 only)


#--------------------------------------------------------------------------------
[IgnoreIndices]
#        Define indices of volumes that are ignored for GLM calculation in the time series.
#        Indices are running from  [1...Timepoints].
#        Indices >Timepoints or <1 are NOT considered for <IArr>.
IgnoreIndicesLines    = 2        #
IgnoreIndicesColumns  = 4        # maximum is 10
IArr[1]= 0 0 0 0
IArr[2]= 0 0 0 0


#------------------------------------------------------------------------------ end
```

```
#-------------------------------------------------------------------------------
# GLM Paradigm Input File <p1short.ini> Josef Pfeuffer Aug-2006
#-------------------------------------------------------------------------------
[Design]
t-threshold       = 4             # [typical: 4]  [range: >=0]  [optional]
StartInlineAtMeas = 20            # [typical: 20] [range: >0]   [optional]
StartingIgnores   = 0             # [typical: 0]  [range: >=0]  [optional]
#HighpassCutOffCycle = 25         # [typical: 20] [range: >0]   [optional]
                                  # switch off highpass-filter with 99999.
#HRFDelay_s        = 0            # [typical: TR/2] [range: +/-TR] [optional]


#-------------------------------------------------------------------------------
[DesignMatrix]
#        Note: always ADD A CONSTANT COLUMN (ie. offset) to model the data correctly.
#        If more measurements occur than lines specified (which is usually the case),
#        the matrix will be used in a cyclic way.

DesignMatrixLines     = 1
DesignMatrixColumns   = 2         # MUST be identical to the other two matrices !!!

#----------- The DesignMatrix
#        Cond1 Offset
DMat[1]= 0     1                  # dummy line: see DesignTMatrix

#-------------------------------------------------------------------------------
[DesignTMatrix]   # [optional] if defined: replaces [DesignMatrix]
#        Note: amplitudes are preset to 0, last covariate is preset to 1 (offset);
#        time base can be: volumeBase (TR)  or timeBase [s]!
#        stimuli are convoluted with HRF and resampled to volume TR for GLM

DesignTimeBase        = 0.1       # [s] - set 0 for volumeBase (TR)
#DesignTimeShift      = 0         # [s] [optional]
DesignTMatrixLines    = 3
DesignTMatrixColumns  = 4         # MUST be (DesignMatrixColumns + 2)

#----------- The DesignTimeMatrix defines time points, time (first volume) starts at ZERO !
#        Onset Dur   Cond1 Offset
DTMat[1]=  0   0.2   50    1      # a brief stimulus with 50% intensity
DTMat[2]= 10   0.1  100    1      #                               100% intensity
DTMat[3]= 40   0     0     1      # end point of paradigm cycle, TR = 2s

#-------------------------------------------------------------------------------
[ContrastMatrix]
#        Note: if you are contrasting conditions against each other,
#        the sum of the contrast-vector should be zero.
ContrastMatrixLines   = 1
ContrastMatrixColumns = 2         # MUST be identical to DesignMatrixColumns

#        Cond1 Offset
CMat[1]= 1     0                  #

#-------------------------------------------------------------------------------
[ConvDerivMatrix]
#        Note: Careful with the derivatives! If you derive a constant column or a box-car
#        shape, the result will not be useful.  Values 0 and 1 only!
#        The last column value should usually be zero (ie. constant offset).
ConvDerivMatrixLines   = 2
ConvDerivMatrixColumns = 2        # MUST be identical to DesignMatrixColumns

#        Cond1 Offset
CDMat[1]= 1    0                  # switch for HRF convolution (0/1 only)
CDMat[2]= 1    0                  # switch to add derivative  (0/1 only)

#-------------------------------------------------------------------------------
[IgnoreIndices]
#        Define indices of volumes that are ignored for GLM calculation in the time series.
#        Indices are running from  [1...Timepoints].
#        Indices >Timepoints or <1 are NOT considered for <IArr>.
IgnoreIndicesLines    = 2         #
IgnoreIndicesColumns  = 4         # maximum is 10

IArr[1]=  0 0 0 0
IArr[2]=  0 0 0 0
#------------------------------------------------------------------------- end
```

7. Important Notes

Protocols created using this package ***cannot be guaranteed to be successfully converted*** when installing a new software version on your scanner. Thus your protocols created with this package may be lost when upgrading.

Independent of the first note, it ***cannot be guaranteed that the functionality*** provided with this package ***enters the standard product*** software releases.

Questions to the user:

- Is the image quality sufficient for clinical cases?
- Did you change the predefined protocols? How?
- Does the GLM evaluation improve your fMRI results?
- Is the input of external design files useful? – Please comment on workflow and usability.

    For what paradigms are you using the ExternalDesign input file?

Is the current extended functionality sufficient to fulfill your needs for ONLINE and OFFLINE fMRI analysis?

- What further improvements are necessary or anticipated?

8. Installation

Installation of this package should be performed by an experienced user. For installation you need to access the file-system of the scanner ("Advanced User" functionality), which requires a password. Please contact the local application support for this information.

Installation can be executed with the following steps:

- Copy the installation ZIP file to the scanner C:\temp

- Click on it with the right mouse button & select "Extract all" - Select „Next"

- Enter c:\ for the target path and click on "Next"

- open Explorer and double-click 'Install_WIP.bat':

       Read the output carefully and check for error-messages.

       A log file will be written to C:\MedCom\log

- Reboot scanner to get all changes into effect.

The following files will automatically be copied to the system:

```
C:\MedCom\MriCustomer\seq\ep2d_glm_383.* / ep2d_DistorCor_383.*
C:\MedCom\MriCustomer\EvaDefProt\BOLD\glm_moco.evp
C:\MedCom\MriCustomer\glm_designs\*.ini
C:\MedCom\bin\IceEPIDistorCor.evp/ .dll
C:\MedCom\MCIR\Med\lib\libIceEPIDistorCor.so
C:\MedCom\MriCustomer\IceConfigurators\IceEPIDistorCor.evp
```

A *fix has to be applied if you get the following error msg (logviewer):*

```
==================== ExtDesignReader: pcFileName =
</opt/medcom/mricustomer/glm_designs/p1cond.ini> =================================
ERROR reading/processing external design file. Detailed error description: ExtDesignReader: ERROR
opening ini-file: p1cond.ini
 status = 0x0
```

       fix to be applied manually on the MRIR:

```
Run: ssh root@mrir
mrir:~ # cd /opt/medcom
mrir:/opt/medcom # ln -s MriCustomer mricustomer
mrir:/opt/medcom # ls -l mricustomer
lrwxrwxrwx 1 root root 11 Nov 10 17:01 mricustomer -> MriCustomer
mrir:/opt/medcom # exit
```

Create new measurement protocols using the Exam-Explorer for the new sequences *ep2d_glm_383* and *ep2d_DistorCor_383*. The new protocols can be inserted into an existing exam by the "insert sequence" functionality from the "User Sequences" tree. You could use the same protocol settings as used with the original ep2d_pace sequence.

Note: Additional options are available on the BOLD2 card.

New evaluation protocols could also be found when using the BOLD-Evaluation platform under the name ADVGLM_*.evp. Those can be used to evaluate existing or new BOLD datasets using a General Linear Model statistics.

Known limitations (EPI 2D DistorCor):

- You must select **Magnitude/Phase** mode on the Contrast tab.
- GLM statistics and Motion correction is only possible **offline**.
- Do not use the APPLY button to start scanning, use SCAN