================
|| S V T MANUAL ||
================

SUB − VOLUME THRESHOLDING (SVT) TECHNIQUE FOR ANALYZING FUNCTIONAL IMAGES

This file explains (step−by−step) the procedures one needs to follow to do the SVT (Sub−Volume Thresholding) statistical Analysis for determining the statistically significant regions of activation in (single or multiple studies of) Functional Data. This technique, is proposed by: I. Dinov, P. Thompson, R. Woods, M. Mega, C. Holmes, DW Sumners, S. Saxena and A. Toga. Implementation, testing and documenting of the new method is done by the same group of researchers.

("jot" could be replaced by any other editor like "vi")
("pico", "edit" etc. to properly edit batch and source files)

I. Single−subject studies

These studies involve determining of statistically significant metabolic changes of a single−subject scanned twice, under baseline (referred to as "Rest") and stimulus (referred to as "Act", for activation) conditions.

Suppose A.img and R.img are the two raw volumes of size X * Y * Z, where X is the fastest−varying index and Z is the slowest−varying index, no matter if the image represents transverse (axial), sagittal or coronal planes.

The following describes the exact order of the steps needed to be performed to get a correct final analysis:

(1) autarch%> csh batch_myFnI3D (this step is only necessary if the files need to be non−trivially Flipped, or intensities Inverted (FnI, Flip & Invert). Sometimes, if the raw images came from other formats (NIH) some of the x_step, y_step or the z_step could be negative (like −1), which means that the dimensions are reversed. You could use a properly selected line of "batch_raw2minc" to convert the raw data file "*.img" into a "*.mnc" (MINC−format) file that you could visualize (using: my_host%> Display file.mnc) to see if you'd need a flip.)

− jot batch_raw2minc, with the proper voxel size, dimensions
    and volume orientation − sagittal, transverse, coronal;
− Display file.mnc, to visually inspect for a need of an FnI,
    Flip & Invert;

– jot batch_myFnI3D, edit properly the batch file, giving the
file name of the volume to be flipped, dimensions etc;
– csh batch_myFnI3D, execute (do) the FnI;
(For many volumes of the same type (protocol) this could
be done simultaneously for all volumes in one pass).

(2) autarch%> csh batch_makeaheader (this would make the appropriate headers of the raw files. You'd need to know the volume dimensions and the voxel–size. These headers are in AIR–format (Mayo Clinic, UK) and are needed by the AIR registration routines.)

– jot batch_makeaheader, edit properly the file. All headers
could be created in one pass only;
– csh batch_makeaheader, after the correct editting is done, execute
the batch_command_file.

(3) autarch%> csh batch_warps (this would do two things: First it would warp the A.img to R.img (registers A to R), call the new resliced volume A2R.img. Secondly, it would warp warp A2R.img and R.img to "avgedit.img" – the reference "Average" anatomical MRI volume associated with the probabilistically defined search regions (In the MNI atlas – caudate, cerebellum, frontal, insula, occipital, parietal, putamen, temporal lobes, thalamus).
Currently, "avgedit.img" contains the average_53_MNI study, which is the core of the currently used probabilistic atlas (ROI's).

Call the new (resliced) volumes A2R2Avg.img and R2Avg.img (Note: This second step could vary a little depending on the study of interest – see "(3)" in Multi–Subject Analysis). Of course, both warping and reslicing are performed at each step separately.

After this stage, the "A2R2Avg.img", "R2Avg.img", "avgedit.img" and "lobes9_diff_intens.img" should all look registered (convert them to *.mnc and visualize using "sgi%> register.opengl file1.mnc file2.mnc"). The file "lobes9_diff_intens.img" contains a mask data for the ROI's – probabilistically defined. 1 <= Caudate <=20 25<=Cerebellum <= 45, 50<=Frontal<=70, 75<=Insula<=95, 100<=Occipital<=120, 125<=Parietal<= 145, 150<=Putamen<=170, 175<=Temporal<195, 200<=Thalamus<=220. These are the corresponding intensities of the mask_file and of course, Higher_intensities mean higher probabilities, lower_intensities mean lower probabilities – of the corresponding voxel being inside the ROI. All the warpings use Roger Woods AIR3.0 auto–image registration tool.)

– jot batch_warps, edit properly the batch file;
– csh batch_warps, execute the batch.

Note: It may be wise to concatenate the field (.air & .warp) files and reslice only once, instead of reslicing twice, since interpolation done twice will degrade the results a little. See batch_warps_combine.

(4) autarch%> csh batch_normalize (this would do the inter−subject normalization. Equalizes the first two moments of the signals, mean & variance. Some studies may require also intra−subject normalization, where, say the intensities of each volume are normalized to these of the "putamen", say, see "intraSubj_normPutamen" and the corresponding batch file. Typically the putamen's and cerebellar intensities are more stable, and rarely show severe atrophies. That is why they can be used as references.

(5) autarch%> csh batch_SVT (this would do the necessary Single−Subject stat_analysis on the difference image (A2R2Avg.img − R2Avg.img). Call the output A_R_SVT.img.

− jot batch_SVT, edit properly the batch file;
− csh batch_SVT, execute the batch.

NOTE: If you ever change the probabilistic ROI atlas you would need to execute: "varEstCorrRandFct1.out" and "extrSurBoxProbStr.out". The first routine determines the necessary CF's (correction factors) for the variance estimates (required for the stat_analysis), based on the shape of the ROI's, see paper. The second one merely finds the smallest bounding box about each ROI and needs to be executed first (speeds up the computations), and the outputs of it need to be fed into all of the following routines. The outputs of "varEstCorrRandFct1.out" also need to be used in all routines that follow.

(6) autarch%> csh batch_raw2minc1 (this final step would convert the A_R_SVT.img to A_R_SVT.mnc (remember to look at both: the positively and negatively statistically significant SVT images).

− jot batch_raw2minc1, edit correctly the batch file;
− csh batch_raw2minc1, execute the batch.

(7) autarch%> register.opengl ./avgedit.mnc ./A_R_SVT.mnc (this would show in register the output of the stat_analysis. Visualize the SVT statistics on top of the anatomical "avgedit.mnc" or the functionl data or overlayed on the probabilistic atlas "lobes9_diff_intens.mnc".

−−

II. Multiple−subject Studies

These studies involve determining of statistically significant metabolic changes for multiple subjects. It can be used for determining variations between groups (like amnestic and memory−retrieval−deficit groups) or for analyzing data for a single group of subjects each scanned twice, under baseline (referred to as "Rest") and stimulus (referred to as "Act", for activation) conditions (like right versus left−handed subjects performing finger−opposition task, for example).

Suppose A(k).img and R(k).img are the Active (or group 1) raw volumes and Rest (or group 2) raw volumes ALL of size X * Y * Z, where X is the fastest−varying index and Z is the slowest−varying index, no matter if the image orientation is transverse (axial), sagittal or coronal.

The following describes the exact order of the steps needed to be performed to get a correct final analysis (this is very similar to part I.):

(1) autarch%> csh batch_myFnI3D (As in I. do flips if need be!)

(2) autarch%> csh batch_makeaheader (As in I. make ALL headers!)

(3) autarch%> csh batch_warps (Warp ALL to one of them, best representing the anatomy of the subjects being studied − as in I. this is the first registration step. Now there are at least three variations for the second warping step. Call the resliced volumes A(k)_2_A*.img, R(k)_2_A*.img, and the corresponding fields A(k)_2_A*.air, R(k)_2_A*.air, where "A*.img" is one of the data sets chosen as a reference volume.

(3.2.1) Warp the resulting resliced volumes to the average: "avgedit.img". Obtain the fields A(k)_2_A*_2_Avgedit.F, R(k)_2_A*_2_Avgedit.F, where "F" stands for "air" (for affine) or "warp" (for non−linear) polynomial registration fields. Reslice using these fields to get the final registered volumes, Figure 3.2.1;

(3.2.2) As in (3.2.1) obtain the fields "A(k)_2_A*_2_Avgedit.F" and "R(k)_2_A*_2_Avgedit.F", but use "combine_air" or "combine_warp" (for combining linear followed by a non−linear fields). Call the combined fileds "A(k)_Dbl_2_Avg.F" and "R(k)_Dbl_2_Avg.F". Finally, reslice using these new fileds, Figure 3.2.2;

(3.2.3) Using the resliced volumes from warping step one find the "average" of the volumes: "A&R_Avgd_2_A*.img". Then register "A&R_Avgd_2_A*_2_Avgedit.img" to "avgedit.img", using an affine or non−affine warp, call the field "A&R_Avgd_2_A*_2_Avgedit.air/warp". To get the final registration of A1 to "avgedit.img" (going through A*.img) combine A1_2_A*.air followed by "A&R_Avgd_2_A*.img" and reslice using the combined field. Similarly for the other A(K) and the R(K) volumes, Figure 3.2.3;

The bottom line is that the second field−to−be−applied is always the same: "A&R_Avgd_2_A*_2_Avgedit.air/warp" − it does not depend on A(k) or (R(k). A variation of (3.2.3) is re−registering "A(k).img" ..., "R(k).img" to "A&R_Avgd_2_A*.img" obtaining new first fields: "A1_2_Avgd.air" etc. and then use these as the first fields into to "combine_air/warp" as in (3.2.3), instead of using "A1_2_A*.air".

(4) autarch%> csh batch_normalize (Normalization, as in I.)

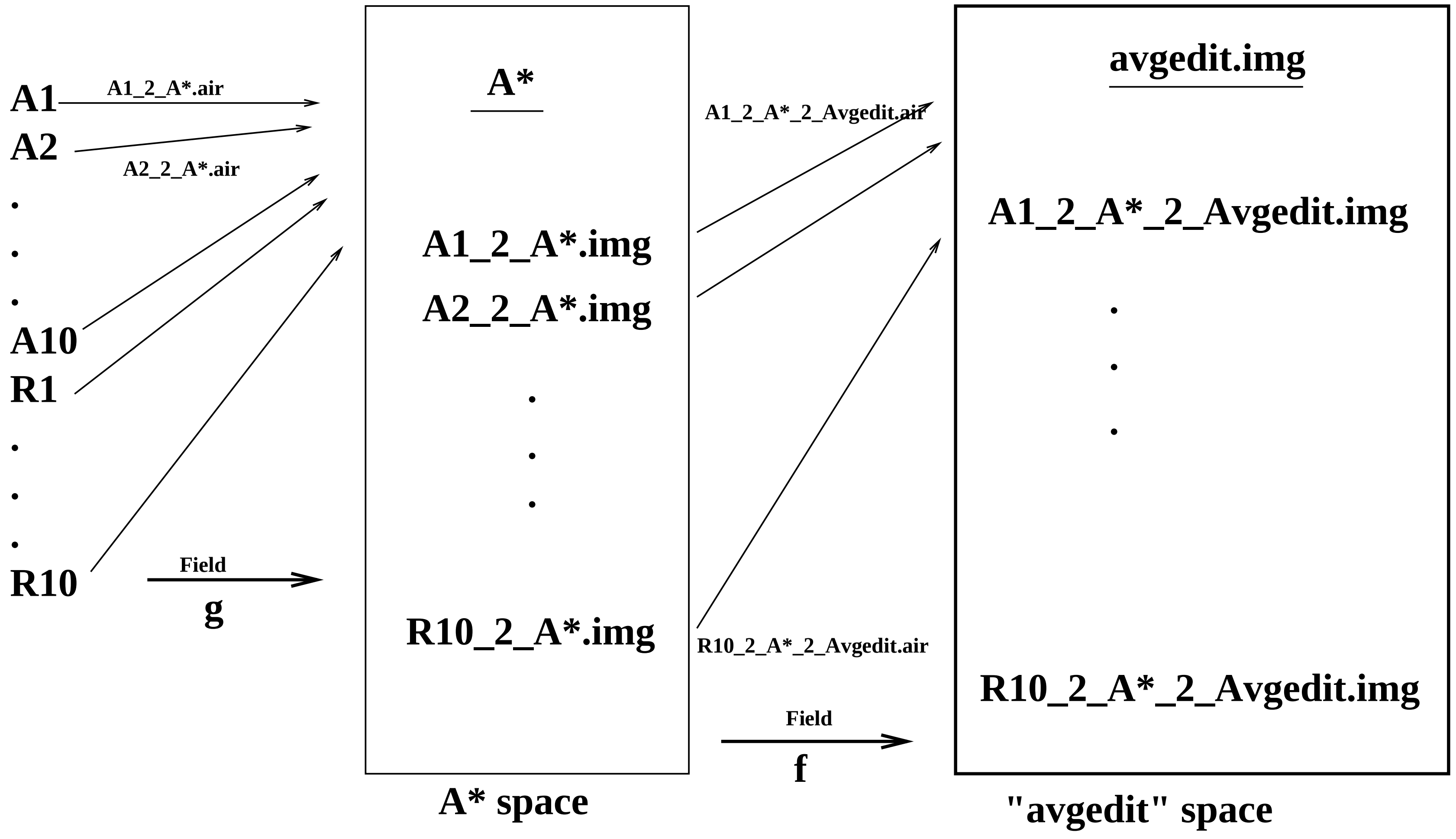
(5) autarch%> csh batch_SVT_MS (To do the Multi_subject_SVT stat_analysis)

(6) autarch%> csh batch_raw2minc1 (To make "*.mnc" file of the result)

(7) autarch%> register.opengl ./avgedit.mnc ./A_R_SVT_MultSubj.mnc (to visualize the results of the statistical analysis, SVT technique.)

NOTE: "autarch" everywhere could be replaced by the name of the SGI host you have logged on. Common SGI hosts are: "terminus", "nessus", "thrax", "vodalus", "fuligin" and "triskele".

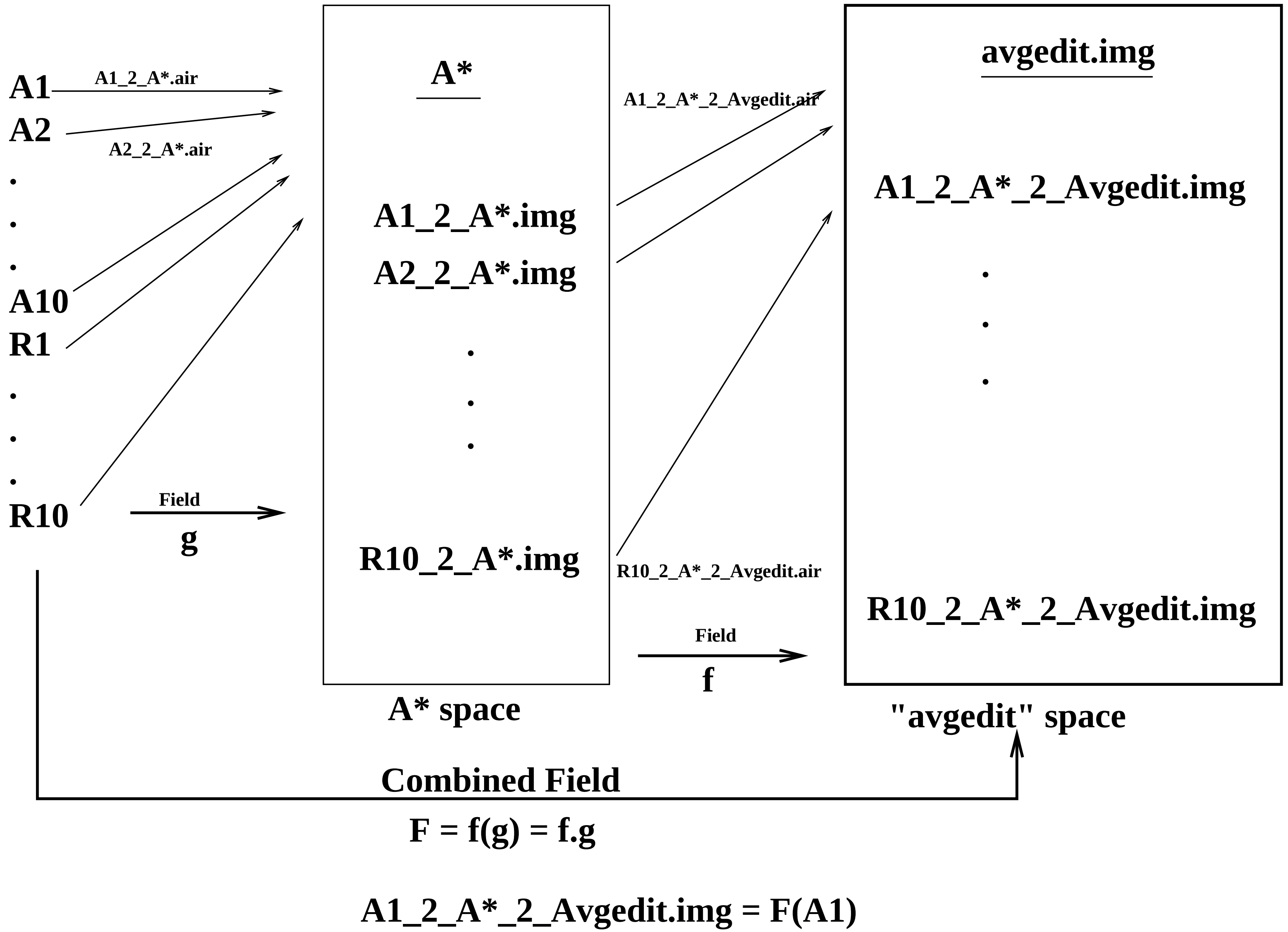
Figure 3.2.1



A1_2_A*.img = g(A1.img)

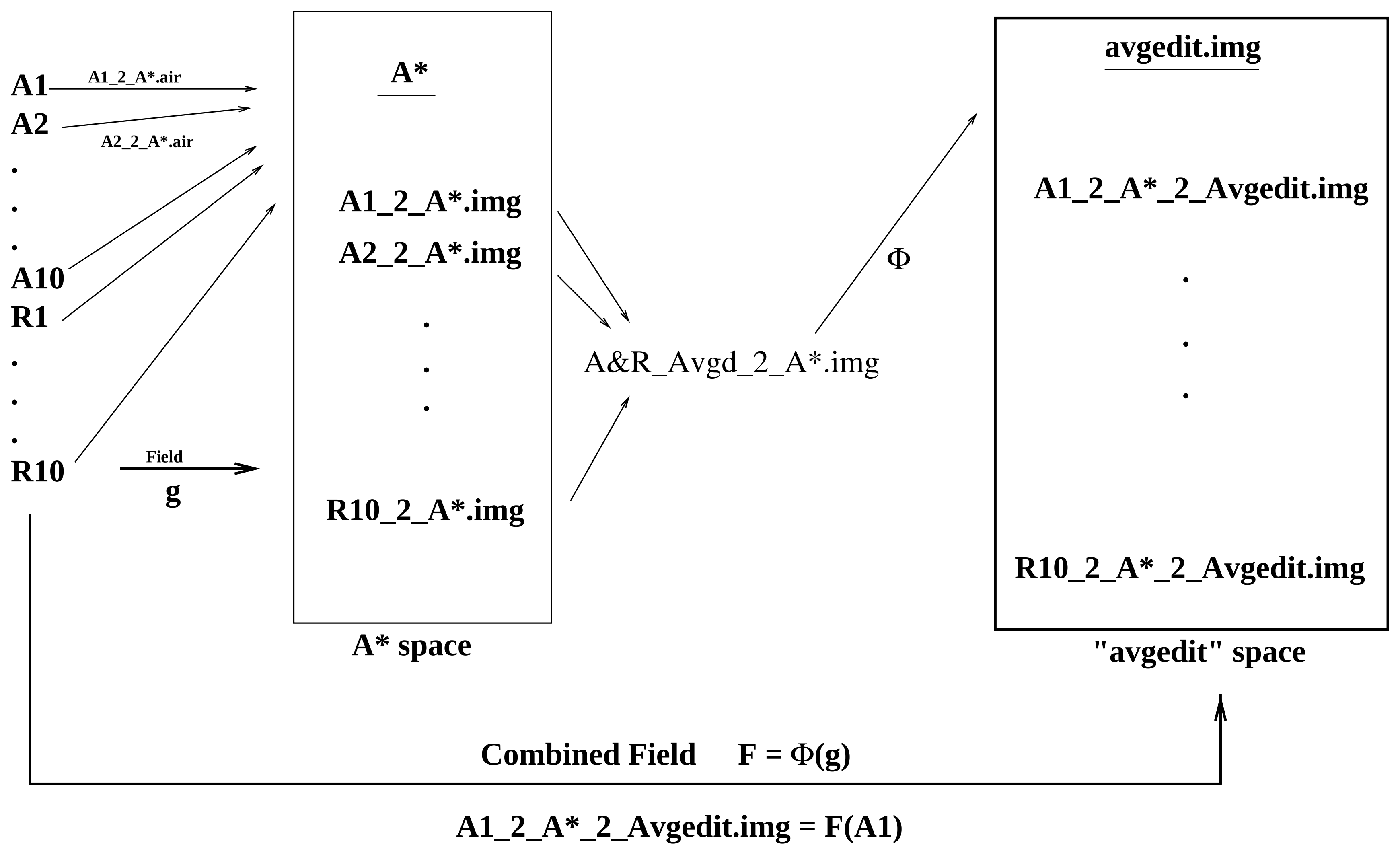
A1_2_A*_2_Avgedit.img = f(A1_2_A*.img)

Figure 3.2.2



A1
A2
.
.
.
A10
R1
.
.
.
R10
A1_2_A*.air
A2_2_A*.air
Field
g
A*
A1_2_A*.img
A2_2_A*.img
.
.
.
R10_2_A*.img
A* space
A1_2_A*_2_Avgedit.air
R10_2_A*_2_Avgedit.air
Field
f
avgedit.img
A1_2_A*_2_Avgedit.img
.
.
.
R10_2_A*_2_Avgedit.img
"avgedit" space
Combined Field
F = f(g) = f.g
A1_2_A*_2_Avgedit.img = F(A1)

Figure 3.2.3



A1
A2
A10
R1
R10
A1_2_A*.air
A2_2_A*.air
Field
g
A*
A1_2_A*.img
A2_2_A*.img
R10_2_A*.img
A* space
A&R_Avgd_2_A*.img
Φ
avgedit.img
A1_2_A*_2_Avgedit.img
R10_2_A*_2_Avgedit.img
"avgedit" space
Combined Field F = Φ(g)
A1_2_A*_2_Avgedit.img = F(A1)