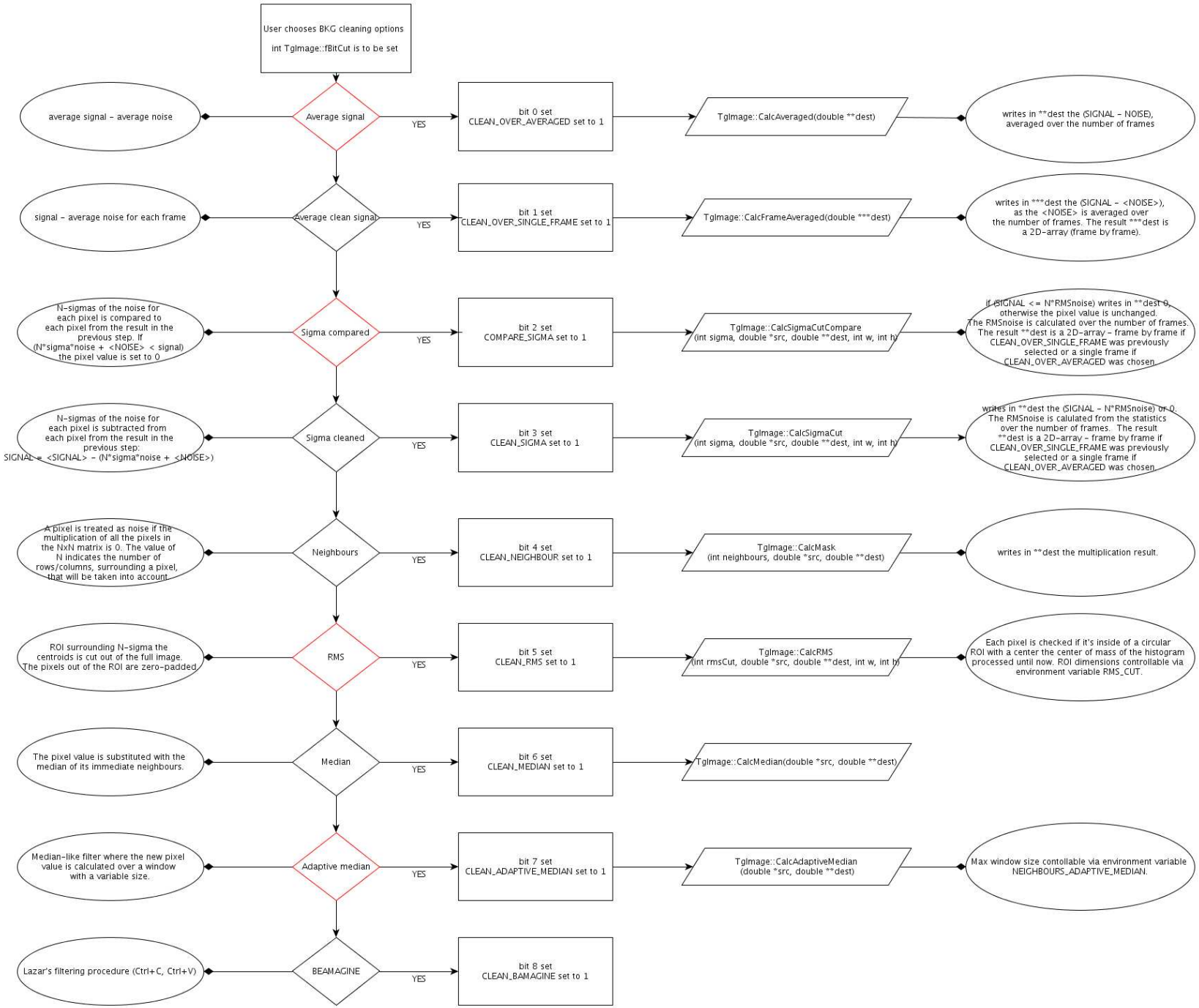
User chooses BKG cleaning options
int TgImage::fBitCut is to be set
average signal - average noise
Average signal
YES
bit 0 set
CLEAN_OVER_AVERAGED set to 1
TgImage::CalcAveraged(double **dest)
writes in **dest the (SIGNAL - NOISE), averaged over the number of frames
signal - average noise for each frame
Average clean signal
YES
bit 1 set
CLEAN_OVER_SINGLE_FRAME set to 1
TgImage::CalcFrameAveraged(double ***dest)
writes in ***dest the (SIGNAL - <NOISE>), as the <NOISE> is averaged over the number of frames. The result ***dest is a 2D-array (frame by frame).
N-sigmas of the noise for each pixel is compared to each pixel from the result in the previous step. If (N*sigma*noise + <NOISE> < signal) the pixel value is set to 0
Sigma compared
YES
bit 2 set
COMPARE_SIGMA set to 1
TgImage::CalcSigmaCutCompare (int sigma, double *src, double **dest, int w, int h)
if (SIGNAL <= N*RMSnoise) writes in **dest 0, otherwise the pixel value is unchanged. The RMSnoise is calculated over the number of frames. The result **dest is a 2D-array - frame by frame if CLEAN_OVER_SINGLE_FRAME was previously selected or a single frame if CLEAN_OVER_AVERAGED was chosen.
N-sigmas of the noise for each pixel is subtracted from each pixel from the result in the previous step: SIGNAL = <SIGNAL> - (N*sigma*noise + <NOISE>)
Sigma cleaned
YES
bit 3 set
CLEAN_SIGMA set to 1
TgImage::CalcSigmaCut (int sigma, double *src, double **dest, int w, int h)
writes in **dest the (SIGNAL - N*RMSnoise) or 0. The RMSnoise is calulated from the statistics over the number of frames. The result **dest is a 2D-array - frame by frame if CLEAN_OVER_SINGLE_FRAME was previously selected or a single frame if CLEAN_OVER_AVERAGED was chosen.
A pixel is treated as noise if the multiplication of all the pixels in the NxN matrix is 0. The value of N indicates the number of rows/columns, surrounding a pixel, that will be taken into account.
Neighbours
YES
bit 4 set
CLEAN_NEIGHBOUR set to 1
TgImage::CalcMask (int neighbours, double *src, double **dest)
writes in **dest the multiplication result.
ROI surrounding N-sigma the centroids is cut out of the full image. The pixels out of the ROI are zero-padded.
RMS
YES
bit 5 set
CLEAN_RMS set to 1
TgImage::CalcRMS (int rmsCut, double *src, double **dest, int w, int h)
Each pixel is checked if it's inside of a circular ROI with a center the center of mass of the histogram processed until now. ROI dimensions controllable via environment variable RMS_CUT.
The pixel value is substituted with the median of its immediate neighbours.
Median
YES
bit 6 set
CLEAN_MEDIAN set to 1
TgImage::CalcMedian(double *src, double **dest)
Median-like filter where the new pixel value is calculated over a window with a variable size.
Adaptive median
YES
bit 7 set
CLEAN_ADAPTIVE_MEDIAN set to 1
TgImage::CalcAdaptiveMedian (double *src, double **dest)
Max window size contollable via environment variable NEIGHBOURS_ADAPTIVE_MEDIAN.
Lazar's filtering procedure (Ctrl+C, Ctrl+V)
BEAMAGINE
YES
bit 8 set
CLEAN_BAMAGINE set to 1