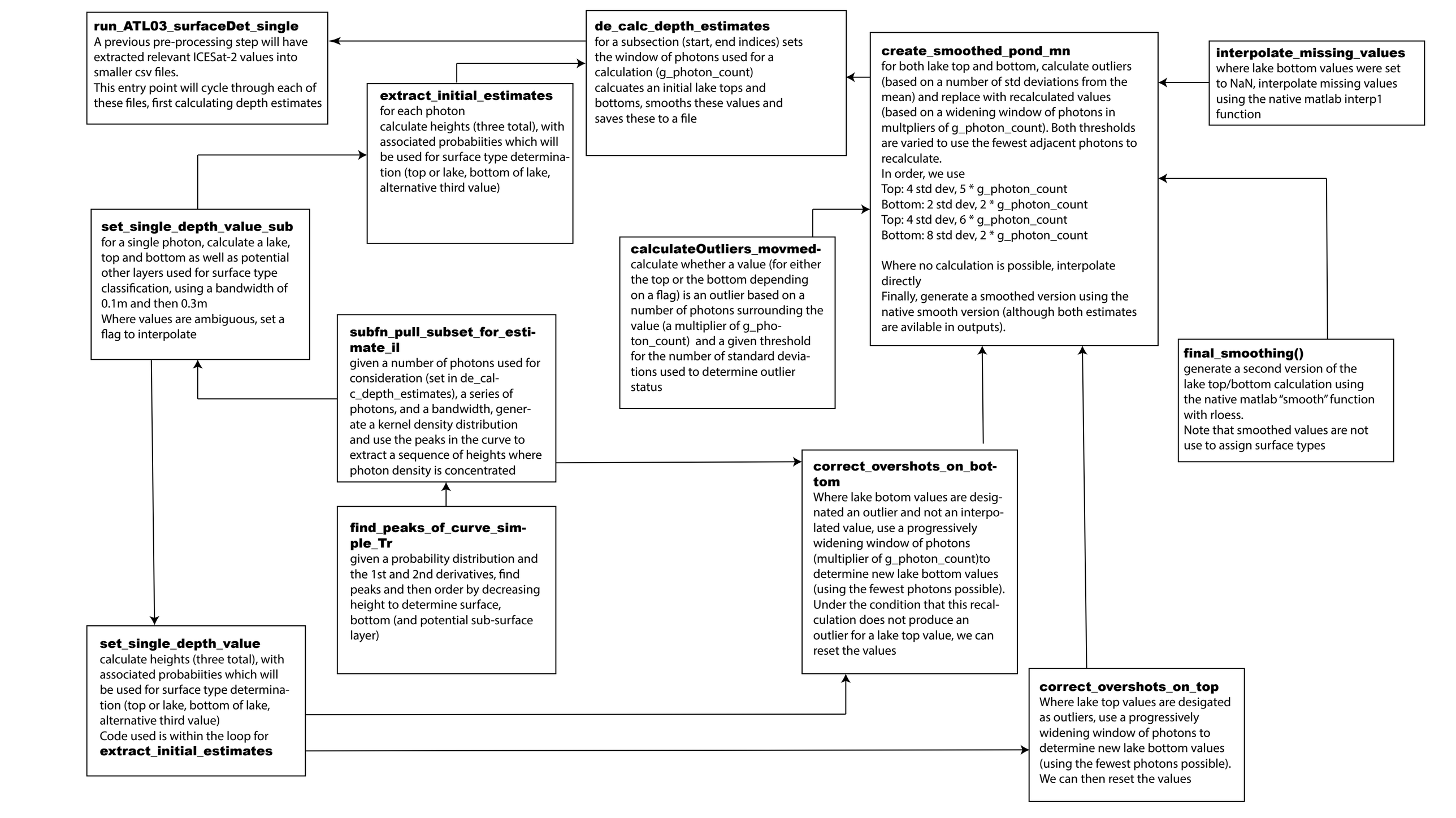
run_ATL03_surfaceDet_single
A previous pre-processing step will have extracted relevant ICESat-2 values into smaller csv files.
This entry point will cycle through each of these files, first calculating depth estimates
de_calc_depth_estimates
for a subsection (start, end indices) sets the window of photons used for a calculation (g_photon_count)
calcuates an initial lake tops and bottoms, smooths these values and saves these to a file
extract_initial_estimates
for each photon
calculate heights (three total), with associated probabiities which will be used for surface type determination (top or lake, bottom of lake, alternative third value)
create_smoothed_pond_mn
for both lake top and bottom, calculate outliers (based on a number of std deviations from the mean) and replace with recalculated values (based on a widening window of photons in multpliers of g_photon_count). Both thresholds are varied to use the fewest adjacent photons to recalculate.
In order, we use
Top: 4 std dev, 5 * g_photon_count
Bottom: 2 std dev, 2 * g_photon_count
Top: 4 std dev, 6 * g_photon_count
Bottom: 8 std dev, 2 * g_photon_count
Where no calculation is possible, interpolate directly
Finally, generate a smoothed version using the native smooth version (although both estimates are avilable in outputs).
interpolate_missing_values
where lake bottom values were set to NaN, interpolate missing values using the native matlab interp1 function
set_single_depth_value_sub
for a single photon, calculate a lake, top and bottom as well as potential other layers used for surface type classification, using a bandwidth of 0.1m and then 0.3m
Where values are ambiguous, set a flag to interpolate
calculateOutliers_movmed-
calculate whether a value (for either the top or the bottom depending on a flag) is an outlier based on a number of photons surrounding the value (a multiplier of g_photon_count) and a given threshold for the number of standard deviations used to determine outlier status
subfn_pull_subset_for_estimate_il
given a number of photons used for consideration (set in de_calc_depth_estimates), a series of photons, and a bandwidth, generate a kernel density distribution and use the peaks in the curve to extract a sequence of heights where photon density is concentrated
final_smoothing()
generate a second version of the lake top/bottom calculation using the native matlab "smooth" function with rloess.
Note that smoothed values are not use to assign surface types
find_peaks_of_curve_simple_Tr
given a probability distribution and the 1st and 2nd derivatives, find peaks and then order by decreasing height to determine surface, bottom (and potential sub-surface layer)
correct_overshots_on_bottom
Where lake botom values are designated an outlier and not an interpolated value, use a progressively widening window of photons (multiplier of g_photon_count)to determine new lake bottom values (using the fewest photons possible). Under the condition that this recalculation does not produce an outlier for a lake top value, we can reset the values
set_single_depth_value
calculate heights (three total), with associated probabiities which will be used for surface type determination (top or lake, bottom of lake, alternative third value)
Code used is within the loop for
extract_initial_estimates
correct_overshots_on_top
Where lake top values are desigated as outliers, use a progressively widening window of photons to determine new lake bottom values (using the fewest photons possible). We can then reset the values