



## LLUVPointExtractor 2 User Guide

130.01.1611.UG

Nov 16, 2016

### LLUVPointExtractor 2 User Guide

LLUVPointExtractor is a command line tool for extracting a vector or vectors from a series of source LLUV files into an output LLUV file with each point time stamped. The resultant output can then be plotted with DiagDisplay or converted to KML or used for your own studies. This tool is most useful when you want to extract velocities over time for a specific location.

Command line tools run in a Terminal window and require some basic unix scripting (bash,perl,...) knowledge.

The output file is a typical LLUV file except that time columns are added to the vectors. These timestamped vector files can be plotted in DiagDisplay or read for your own analytical usage. LLUVtoKML can convert these to KML with vector time so that time controls in GoogleEarth will animate the velocities. Note, GoogleEarth typically shows the timestamp converted to local time; this can be changed. Also note, all other current SeaSonde LLUV Readers will support this format but will typically ignore the timestamp; for Instance, SeaDisplay will plot all the vectors on top of each other.

The files to scan can be specified as:

1. A folder with a time span starting from last alphabetical file name.
2. A file within a folder to start from with an optional time span.
3. A text file that contains a list of pathed files to merge.
4. One or more files.

When LLUVPointExtractor is run from the command line without any options (or with '-h' or '-help' option) it will list all the options that can be set. Each option needs to be separated by at least one space. Options that require a string or file path that might contain spaces can be wrapped in quotes (example -input="/Test Folder/")

Usage: LLUVPointExtractor [-option[=value]]

-diag=<verbocity>      -- Output diagnostics to stdout. <verbocity>  
                          0=quiet,1000=all.

```

-plist=<file/folder> -- Plist file to use or folder containing
                      LLLUVPointExtractor.plist
-source=<file/folder> -- LLUV file or folder containing LLUV to read.
-input=<file/folder>  -- same as -source.
-output=<file/folder> -- LLUV file output to append or folder to write
                      to.
-suffix=<suffix>     -- Addition suffix to add to radial output
                      filename.
-method=<type>        -- Maximum Minimum Largest Smallest Closest All
                      Average Median Neighbor FourNeighbors.
-distance=<km>        -- Use vectors found within this kilometer
                      radius.
-latlon=<lat>,<lon>   -- Location of points. Wide variety of lat,lon
                      formats accepted.
-xy=<x>,<y>           -- easting,northing kilometer location from -
                      latlon or origin of source.
-rb=<r>,<b>           -- range kilometer at bearing True from -latlon
                      or origin or source.
-append=Yes/No        -- If No, delete output file before adding
                      vectors. Default is Yes.
-tables=Yes/No        -- If Yes, new vectors will be added as separate
                      LLUV table. Default is No.
-angwidth=<deg>      -- If not zero, use vectors found within annulus
                      segment of distance and this angular width.

```

**Option -diag** controls diagnostic output messages.

**-diag**=<verbocity> where <verbocity> is a number 0 to 1000. Sets how much information LLUVMerger outputs to stdout during processing. 0=quiet, 4=general information, 1000=most detailed. Set this to a high value to help debug LLUVMerger. Set to 0, so that only failure messages are output.

If you set to diag to 100 or greater it will output the source velocities for each vector being merged. You can save this by redirecting the standard output to a file (append " >/MergeInfo.txt" to your command to create the file MergeInfo.txt onto the root of the drive)

**Options -source -input** specify input file or folder containing LLUV files.

Only valid LLUV files will be used.

**-source**=<file/folder> is a path to source folder or file. If a file then only that file is processed. If a folder is then all LLUV files in the folder will be processed.

**-input**=<file/folder> is same as -source

**Options** **-output** and **-suffix** control the output folder and filename.

**-output**=<folder/file> is a path to the output folder or file. If a folder is used, then a standard LLUV filename is generated using 't' as the LLUV processing type (fourth character in filename) and a weekly timestamp starting on Sunday from the source files. This means that source data which crosses Saturday midnight will generate multiple output files.

**-suffix**=<suffix> will insert <suffix> into the filename before it's normal extension.

**Options** **-plist** **-latlon** **-xy** and **-rb** specify what point(s) to extract.

The search point for extraction is the lat,lon location to search for point(s) to extract within the **-distance** radius circle. It can be specified as a single or multiple points from a property list file or a single point from command line options.

The search point starts at the source origin or specified lat,lon. Then optionally a X,Y or range,bearing distance can be calculated from there. The calculation math uses GPS84 with great circle to figure where the search point is (X,Y is first converted to range,bearing and applied.)

The search point is the origin lat,lon location to search for point(s) to extract. It can be specified as a single or multiple points from a property list file or a single point from command line options.

The search point starts at the source origin or specified lat,lon. Then optionally a X,Y or range,bearing distance can be calculated from there. The calculation math uses GPS84 with great circle to figure where the search point is (X,Y is first converted to range,bearing and applied.)

If you specify a **-plist** option, then options **-latlon** **-xy** and **-rb** will be ignored.

**-plist**=<file/folder> property list file to use or a folder containing file "LLLUVPointExtractor.plist"

The property list must contain a 'Points' Array in the Root dictionary. Each Item in 'Points' Array is a dictionary of the following keywords

**Location\_LatLon**      **string**      <loc>      optional where  
<loc> is lat,lon search location for point to extract. If you do not supply this key the source origin will be used.

**Location\_XY**                    **string** <x,y>                    optional where <x,y> is a easting,northing distance in km from the <loc> or source origin to search for extraction point.

**Location\_RangeBearingstring** <r,b> km where <r,b> is a range km and bearing True from the Location\_XY if specified and the <loc> or source origin for extraction point.

**Method**                            **string**                    <type>                    optional where type is one of Maximum/Minimum/Largest/ Smallest/Closest/All/Average/Median/Neighbor/FourNeighbors. See -method below for explanation of types.

**Suffix**                            **string** <suffix>                    optional suffix to add to output LLUV filename.

If you specify a **-plist** option, then options **-latlon -xy** and **-rb** will be ignored.

**-latlon**=<loc> is optional lat,lon location of the search point.

**-xy**=<x,y> are an x easting and y northing distance from the source origin or -latlon location to specify the search point.

**-rb**=<r,b> are a range km and bearing True from the optional -xy and the source origin or -latlon location to specify the search point.

If -xy and -rb are specified -xy is performed first.

Options **-distance** controls radius size of circle around search point.

**-distance**=<km> look for all vectors within this search radius in km around the search point.

Options **-method** controls how the found vectors are processed.

**-method**=<type> controls how vectors within -closeness distance are merged to one output vector. <type> is one of the following Maximum, Minimum, Largest Smallest, Closest, All, Average, Median, Neighbor, FourNeighbors

**maximum** output the maximum velocity vector within the search area.

**minimum** output minimum velocity vector within the search area.

**largest** output the maximum absolute velocity vector within the search area.

**smallest** output the minimum absolute velocity vector within the search area.

**closest** output the closest vector to the search point location.

**all** output all vectors within the search area.

**average** output average velocity of all vectors within the search area.

**median** output median velocity of all vectors within the search area.

**neighbor** (Not yet implemented) output closest vector not in search area.

**fourneighbors** (Not yet implemented) output four closest vectors not in search area.

Options **-append** and **-tables** controls how vectors are added to the output file.

**-append**=Yes/No controls how vectors are added to the output file. If Yes, vectors are appended if output file already exists. If No, the output file is created each time. The default is Yes which is more suited to processing a single file at a time.

**-tables**=Yes/No controls how vectors are added to the output file. If Yes, for each new file or **-append=yes** for single file, output vectors will enclose each new set in it's own table. Note that much older versions of LLUV Readers will only read the first table.

## **Revision History**

First Draft Nov 11, 2016

## **Copyright and Disclaimer**

This document is copyrighted(c) by CODAR Ocean Sensors, Ltd and cannot be copied or reproduced in all or partial without expressed written consent by CODAR Ocean Sensors, Ltd.