

**Release 2.22 – Revised 8/2/2015 3:13 PM**

# Introduction

This toolkit exists to facilitate quick processing of images outputted from a plenoptic camera. Original developers are Jeffrey Bolan and Kyle Johnson. Current primary developer is Elise Munz. Many thanks to Dr. Brian Thurow, Tim Fahringer, Paul Anglin, Dominic Hildebrandt, and Chelsea Thomason for extensive discussions and contributions that made this toolkit possible. This toolkit was developed in MATLAB R2013b for use in Microsoft Windows. **Hexagonal support is limited to MATLAB R2013a or newer**, while rectangular camera array compatibility includes R2009a and newer. Macintosh compatibility is officially unsupported (although it may be useable with some modifications). Possible issues may arise on MATLAB 2014b and newer. Contact Brian Thurow ([thurow@auburn.edu](mailto:thurow@auburn.edu)) with any questions, comments, or concerns regarding the MATLAB version of this toolkit.

# Change Log

**V2.22 = General Maintenance**

* General cleanup of code/commenting.
* Addition of option for user defined camera parameters

**V2.21 = MAJOR Release.**

* Fixed bug in filtered refocusing.
* See V2.20 changes

**V2.20 = MAJOR Release (BETA).**

* Added new refocusing algorithms:
  + Additive (original/old algorithm).
  + Multiplicative
  + Filtered
* Added option for constant magnification refocusing which uses a user defined volume size and location to produce an orthographic view of the in focus plane of the volume.
* Restructured GUI to include new refocusing options and condense other buttons, all existing capabilities remain.
* Added new batch mode arguments corresponding to new refocusing options, see documentation PDF for more details.

**V2.11 = BUGFIXES.**

* Fixed a (u,v) supersampling incompatibility in refocus.m for MATLAB 2011b and 2012a regarding griddedInterpolant and extrapolation values.
* Tweaked boundary check condition for rectangular calibration to prevent selection of points too near the edge (calrect.m).

**V2.10 = Improved Hexagonal Calibration.**

* Added a new primary hexagonal calibration algorithm which is much faster (seconds vs. minutes) than the previous algorithm.
  + The old algorithm is still available as a fallback method.
* Implemented a calibration menu for the hexagonal case; if the initial calibration is rejected, the user can choose what algorithm to run.
* Added a post-calibration menu for the hexagonal case to improve the user experience and streamline the calibration process. Pay attention to the command line output.

**V2.09 = General Maintenance.**

* Removed deprecated functions from the toolkit.
* Updated toolkitpathv2.m to limit scope when searching for local LFITv2 subfolder.
* Added input argument to imageavg.m to allow setting of averaged file name.
* Added new loadFlag case to computecaldata to permit direct loading of calibration points in text files.
  + Added corresponding GUI option as well.
* Tweaked edge buffer in calrect.m to prevent selection of microlenses too close to the right edge of an image.

**V2.08 = HOTFIX.**

Failed to include normalization updates in previous version; fixed here in v2.08.

**V2.07 = BUGFIXES.**

Bug fixes and other modifications:

* Fixed bug in where the normalization equation used in the toolkit was implemented incorrectly.
* Modified interpimage2.m to window out adjacent microlens data for hexagonal cameras (assumes a circular aperture).
* Optimized refocus.m to perform faster refocusing when using a circular mask.
* Limited search for lastrun.cfg and lastGUI.gcfg to the current directory to limit scope and prevent errors. However, in the event of a loading error due to a corrupted file, delete any lastrun.cfg or lastGUI.gcfg files in the current directory.
* Minor cosmetic update to GUI refocusing schematic diagram.
* Minor updates to documentation PDF.

**V2.05 = Development (BETA)**

New features:

* Added new option to set contrast on a stack basis (as in focal stack generation) to animaterefocus.m (imadjustflag=2) with corresponding GUI options for video and GIF exports.

Other modifications:

* Minor optimizations applied to refocusing movie and focal stack generation functions by preallocating matrices.

**V2.04 = HOTFIX (BETA).**

Bug fixes and other modifications:

* Fixed bug in GUI where when loading default values, the hexagonal camera button would be checked while internally the rectangular (16 MP) camera variables were used. This is manifested in a crash of the calibration function.
* Fixed bug in refocus.m where the circular mask was 1 pixel in radius too large leading to bleeding of data between adjacent microlenses.
* Fixed bug in perspective.m for s,t or no supersampling case.
* Tentatively fixed bug in definition of s and t ranges, affecting in particular the resampling grid for the hexagonal camera.

**V2.03 = MAJOR BUGFIXES.**

Bug fixes and other modifications:

* Changed coordinate system definitions for (u,v,s,t).
* Fixed bug in interpimage2.m which caused a uniform 1 pixel shift in the interpolation step.
* Fixed bug in interpimage2.m where the known and desired coordinate vectors were swapped in the interpolation step.
* Fixed bug in hexagonal resampling where the rectilinear grid had different spacing in s versus t; that is, non-uniform spacing.
* Overhauled perspective.m and refocus.m functions.
  + Updated to reflect new coordinate system definition.
  + Cleaned up old sections of code to be clearer and more consistent.
  + Optimized evaluation of non-integer (u,v) pixel coordinates or supersampled (s,t) for perspective shifts.
  + Fixed bug where refocusing was not shift-invariant.
    - Alpha values for a given focal plane will be slightly different now.
  + Fixed bugs in both functions where (u,v) coordinates were inconsistent and not represented in millimeters.
  + Fixed (u,v) supersampling for refocusing such that a positivity constraint is enforced to zero out negative values and also disabled extrapolation which caused edge artifacts.
* Modified hexagonal resampling to use actual coordinate locations of microlens centers from the calibration step to represent the known coordinate locations associated with the intensity data. Previously a quasi-ideal hexagonal grid was assumed.
* Increased size of image data extraction behind each microlens by 1 pixel on each side.
* Changed (u,v) interpolation to ‘linear’ rather than ‘spline’.
* Rewrote computestrange.m to now accurately calculate the s and t values for the microlenses regardless of the location of the starting calibration point relative to the image center.
* Updated computestrange.m to account for differences between microlens pitch in x and y for hexagonal camera arrays. Also incorporated more exact values of microlens pitch for both camera types.
* Updated documentation to match MATLAB array indexing of (row, column).
* Renamed uvstMatrix to radArray to more appropriately represent the array of intensities (aka radiance).
  + radArray is indexed as radArray(i,j,k,l) where i,j,k,l represent indices corresponding to exact coordinates in uVector, vVector, sVector, and tVector respectively.
* Fixed bug in animaterefocus.m where without Enhance Contrast checked, output would be white due to incorrect limits.
* Fixed bug in animaterefocus.m where it would sometimes crash due to an incorrectly set total number of frames.
* Changed GUI default settings to reflect typical use cases.
* Updated tooltips in the GUI panels.
* Improved error handling in imageavg.m

**V2.01 = BUGFIXES.** Primarily bugfixes in this release.

Fixed bugs:

* Fixed animateperspective.m function where it would crash due to a missing input argument.
* Fixed bug in perspective.m where it would sometimes crash (the function griddedInterpolant due to having a variable as double rather than single).
* Fixed bug in GUI call to animaterefocus.m (the incorrect colormap was passed to the function).
* Fixed tooltip for magnification (magnification in the GUI depends only on camera selection and reference/ruler height).
* Fixed bug in genrefocus.m where it wouldn’t save output in batch mode for any images after the first.
* Fixed bug in Single Image (GUI) mode where loading a new image didn’t actually load a new image but instead reprocessed the old data again.
* Fixed bug in “Clear Calibration…” data option on initial preprocessing interface where the program crashed.
* Changed the sensor height from 24.2 mm to 24.272 mm for the hexagonal array to match the rectangular camera; this will have a small effect on the magnification (and thus the alpha parameter).

New features:

* Added output argument to genfocalstack to permit export of the generated focal stack to the MATLAB workspace.

**V2.00 = MAJOR RELEASE.** Fresh release for a new year. Now with GUI frontend and several bugfixes. New features:

* Cannow place the LFITv2 folder in the directory containing the main demo script and the new toolkitpathv2 function will find and use it before checking the MATLAB user directory.
* Wrote a basic GUI frontend for configuring basic program parameters.
* Can auto load/save calibration data. Select the option in the GUI and run the program. The program will look for a calibration file in the calibration directory with the given image set name (as defined in the GUI). If it finds one, it loads the calibration data (microlens centers). If not, it computes one and saves it.
* Added directory flag to refocusgen.m. 0 will save refocused images into subfolders on a per-image basis and 1 will save refocused images into subfolders depending on the alpha value (as in PSF calculations).
* Added aperture flag to refocusgen.m. 0 will use full aperture behind each microlens. 1 will enforce a circular mask on each microlens (useful for PSF generation and hexagonal processing).
* Wrote GUI interface for processing single plenoptic images.
* Fixed several major bugs from the last official release of the LFI Toolkit v1.06:
  + Fixed critical bug in the calculation of alpha, the depth parameter.
  + Fixed bugs in internal units by converting all internal physical calculations to units of millimeters.
  + Fixed contrast scaling bug in perspective functions.
  + Fixed perspective shift bug in internal betas following v1.06.
  + Fixed image interpolation bug that resulted in negative values which made black backgrounds look gray.
  + Fixed a bug in refocus.m (u,v) supersampling.

# Known Bugs and Development Notes

* BUG: Cinepak codec on MATLAB 2009a for grayscale doesn’t render correctly.
* TODO: Put more user-protection/error-catching into the GUI. For example, catch data entry of (u,v) values that are out of range.
  + Also, invalid TIFF files will cause calibration averaging to crash. Make more user friendly.
* TODO: There is significant room for performance optimization in LFITv2. Specific areas to investigate:
  + The image interpolation process also takes a significant amount of time, but it is unclear how much optimization can be done while still in MATLAB.
* OBSERVATION: JPEG export is hard-coded here to a quality of 90, but in testing, MATLAB’s JPEG compressor appears to be not very good; even a quality setting of 100 yields an image that clearly has been compressed in a lossy fashion. While it’s true JPEG is a lossy compression scheme, the compression artifacts appear to be significantly greater than a comparable JPEG image exported from a commercial graphics program would produce.

END OF DOCUMENT.