

What's New in *Sinemtf*

version 6.6

web posting: May 2016

- Corner density patches in the target reference file can now be reduced in size, to avoid platen corner effects (such as shadows).
- Fixed small (low impact) bug in in the PPI computation that has been present since 2004.

version 6.5

web posting: December 2014

- Changed debug image to:
 - colorize the debug markings,
 - outline the full area for each pattern block (dashed green),
 - outline the test area for each density block with orange line (don't fill the interior)
 - outline the potential test area for each sine/bar block with orange line
 - draw light blue line where the sine/bar peak/valley tests are completed
- Added array bound checking for modulation arrays.
- When TIFF is not available, output debug image is written as BMP (with color).

version 6.4.02

web posting: December 2013

- Altered compilation to use an external TIFF library, and compile with plain gcc.
- Extended PGM reader to handle files with height/width dimensions on separate lines.

version 6.4

web posting: November 2007

- Fixed bug in calculation of 1000ppi spec CTF (option "k").

version 6.3.01

web posting: December 2006

- Fixed bug that erroneously stopped program if target modulation was set to exactly 1.0 in input data file.

version 6.3

web posting: November 2006

- The input values of a nonlinear input/output curve for bar target are no longer restricted to 0 to 1.0 range.
- Added PIV spec checking via menu option j (for 500ppi images).
- Appendix F spec checking now requires invoking menu option k (no spec checking occurs if neither j nor k option is selected).
- Some minor revisions to printout.

version 6.2

web posting: August 2006

- Added option to process bar target with nonlinear input/output curve (I/O data added to end of input target file)
- Added capability to read-in PGM format images (in addition to raw & TIFF)

version 6.1

web posting: November 2005

- Summary data on aliasing results added to verbose output mode (option e).

version 6.0

web posting: June 2005

- New option “g” added to compute digital camera MTF; user inputs lens focal length & target distance, or scale; if color image is output by camera it must be converted to grayscale before input to program.
- Small bug fixed, wherein some curve fits failed if a density patch was completely uniform.

version 5.82

web posting: May 2005

- The diagnostic image (menu option d) is now output in TIFF or PGM format (used to be raw format); TIFF if the TIFF libraries were compiled with the code, otherwise PGM.

version 5.81

web posting: April 2005

- Fixed bug in logistics ‘S’ curvefit printout (verbose mode).
- Fixed bug in MTF spec test at 1.00 cy/mm.
- Cleaned-up code and made compatible with Metrowerks CodeWarrior-for-Mac v9.

version 5.8

web posting: October 2004

- Alias check no longer performed on sine or bar pattern rows that have fewer than 2 periods of data.
- The computed effective frequency (skew angle dependent) is now used in place of input target frequency for low frequency limit of alias checks, for spec bound computation, and in verbose output, where applicable.
- MTF spec bounds now computed via formula (using effective frequency) rather than table lookup.
- MTF/CTF not checked against spec values if skew is greater than 3 degrees.
- General code cleanup: old, commented-out code totally removed.

version 5.7

web posting: August 2004

- Alias detection test is now applied to all sample sets available in a given frequency pattern; aliasing is now flagged if only one set detects aliasing, which is reported out as upscaling aliasing, decimation aliasing, or both types of aliasing being present (some bugs in flagging aliasing also worked-out). Detailed, clarified information on each alias measurement case is now included in MTFOUT file when running verbose output mode (option e). Threshold level for flagging decimation aliasing at frequency marginally above Nyquist increased to 0.45 (threshold level below Nyquist remains at 0.27).
- Expanded the acceptable ordering of gray patch data in input target data file.
- The CTF computed from the image of a bar target is now flagged if it does not meet the IQS spec CTF at 500 or 1000 ppi. This test is only applied if the target modulation, as listed at each frequency in the data file, is less than 1.0.
- Verbose output no longer prints to screen display (is in MTFOUT file).
- Fixed some minor labeling bugs.

version 5.51

web posting: April 2004

- Added check for aliasing if target frequency is marginally beyond computed image Nyquist frequency, e.g., 10 cy/mm sinewave frequency is now checked if image Nyquist is 9.84 cy/mm (500 ppi); test check value increased by 10%, compared to test check value below Nyquist.

- Added defines to printout aliasing computations; see “debug_peaks” in mtf.h (these are normally commented-out).
- Added notes for bar target processing in Help (option r).

version 5.5

web posting: March 2004

- Simplified, more robust bar target processing: program now assumes bar target image when “b” label replaces “s” label in frequency data sets in target data file; density patches are no longer needed and output/input conversion is effectively skipped (previous required digitally adding gray patches to image); image can be in any orientation (user inputs quadrant orientation at runtime); ; all program output labels are changed from MTF & sine (for sine target) to CTF & bar (for bar target); image polarity detection is performed by sampling lowest frequency black bar and adjacent white space; comments in target data file beginning with ## are lifted by program and displayed/printed at runtime; revised BarTgt_Guide document.
- Sanity checks are applied to target data file: patch x,y measurements are compared to stated target width and height (report output if patch measurements exceed stated target dimensions); program stops if minimum and maximum density patches have same value; program now stops right after target data filename is entered, if the file cannot be read.
- Several bugs fixed: rare case of possible (false) zero modulation calculation fixed; linear regression now works on fewer than 5 density patches; alias detection/reporting now occurs even when option “n” is selected; improved logic for identifying & averaging density patches with ‘same’ densities (searches for single 4-patch set, if not found, searches for single 2-patch set); cleaned up compiler warnings in several files.

version 5.0

web posting: January 2004

- Corner coordinate refinement - at user’s option, program will now automatically locate the ‘exact’ reference corners (and use them in computations), given user’s approximate input coordinates for the 3 reference corners. The user’s input coordinates must be within $\pm 15\%$ of the gray patch width to the exact corner and accuracy of auto corner location is dependent on noise level. The auto selected corners are shown in the diagnostic image and are printed to the output file.
- S curve fit - added logistics curve-of-growth to revised curve fit options, for fitting input target reflectance versus output gray level curve; this ‘S’ shaped curve is particularly useful for printer MTF assessment (default curve for that case). See Menu/Help for details on new curve fit options.
- Labeling sequence of the density patches, as supplied by the sine target manufacturer, are now accepted by the program ‘as is’ (no need to rearrange into program’s unique sequencing).
- Bug fixed in logic for selection of number of sinewave rows to average, based on computed skew angle (previously, skew angles of more than 1 degree were not processed as anticipated).
- Image orientation is determined at earlier stage in computations, leading to more refined computation of skew and ppi, and then outer corner coordinates using the skew, ppi, and known, actual density patch dimensions.
- Menu display revised; Help display expanded.

version 4.52

web posting: February 2002

Fixed bug in generation of diagnostic image (d option), occurring when input image is 16bpp. Diagnostic output image is now always set to 8bpp.

version 4.51

web posting: January 2002

Syntactical code modifications were made to allow compiling for Macintosh and Windows using Metrowerks' CodeWarrior-for-Mac-OS-Professional (v7.1). *Sinemtf* algorithm and computations are the same as v4.5.

version 4.5

web posting: November 2001

- Added capability to read-in & process 16 bit/pixel gray level images; the 16 bpp image must be in TIFF format, whereas 8 bpp images can be in raw or TIFF format. If image has 9 to 16 bpp, program assumes 2 bytes per pixel (so it can't handle, e.g., two 12 bpp pixels stored in 3 bytes); also, program cannot handle 16 bpp stored as signed integer.
- Curve fitting options for target reflectance vs. image gray relationship have been expanded, to better handle those cases where a true linear input/output relation does not exist. The user can either select a curve fit type, or let the program select what it thinks is the best curve fit type. The options are:

Linear regression (single straight line of best fit over all data points).

Smoothed spline (local cubic splines that go through smoothed version of data points).

Unsmoothed spline (local cubic splines that go through all data points)

If unsmoothed spline is selected by program or user but the fit fails, it reverts to a piecewise linear fit (local straight lines between adjacent data points).

If the input/output relation is nonlinear, it is best to plot the relation to see if the user-selected or program-selected curve fit type is indeed the 'best' fit.

- If a target has equi-density patches, these are averaged together before the curve fitting, so that the relative weights between all of the data points are equalized. In a sine target, 2, 3, or 4 density patches may have nearly the same density, by design. Density patches are averaged with the following logic (see R_RATIO in mtf.h):

less than 17 target density patches: patches averaged if within 0.040 density units of each other, more than 16 target density patches: patches averaged if within 0.008 density units of each other.

Because of this pre-averaging of equi-density patches, the linear regression curve, and thus the computed MTF, may be slightly different from that obtained with previous *sinemtf* code versions run on the same image.

- For piecewise linear fit, a sine peak or valley gray level that is beyond endpoint of I/O curve is converted to highest or lowest target reflectance patch value, instead of highest or lowest image gray level patch value (logic in previous versions: see this revisions list, v4.1, item 1).

- Added a semi-graphic to menu display as visual aid in selecting correct target orientation.

- New flyer removal logic: sine peak is thrown out as 'noise', if it's gray value is greater than the highest single gray level pixel in the white density patch.

- Added option for intentionally reversing polarity of read-in image before processing; useful with raw images that have 'wrong' endianness for given computer.

- Added clearer, more meaningful error messages.

- Made verbose output clearer, more concise.

version 4.2

web posting: April 2001

- Previous v4.0 & v4.1 quit if the whitest gray patch in the target did not correspond to the computed whitest gray patch in the image. This noncorrespondence can sometimes happen, e.g., in printer MTF assessment. Version 4.2 does not quit under this circumstance and computed MTF most likely is valid, but the following msg is displayed to warn the user that either this has occurred, or some other, real problem has occurred:

IMAGE POLARITY MAY BE WRONG

[but] if above ImageGray values are correct there may not be a problem, otherwise, try re-running with Reversed Polarity image.

- Assumption in previous versions was that the outer vertical edges of the left & right end sine patterns and left & right end gray patches are colinear on the target; which is true for sine target models M6, M13 & M15, but is not true for sine target models M5, M14, M19 (possibly others). Previous program versions did not work correctly for these ‘noncolinear’ targets; this problem has been fixed in v4.2. A slight alteration in constructing the input target data file for M5, M14, M19 is required for this to work, as follows:

Referring to figure C-1 in MTR-96B0000025 document, the target’s origin (0,0 in mm) remains at upper left corner of leftside,topside gray patch; therefore, since the leftedges of the leftmost sine patterns in M5, M14, M19 extend to the left of this origin, these end sine patterns now have NEGATIVE VALUES for their respective x-axis locations (in mm) in the data file. The total target width remains as before: outer edge of leftmost gray patch-to-outer edge of rightmost gray patch, even though the width, if measured across the sine patterns, is greater.

- The logic used for finding valid peaks and valleys in the sine wave was refined. In some cases this may result in a slightly different (i.e. more accurate) modulation compared to earlier versions.

version 4.1

web posting: September 2000

- Removed extrapolation beyond gray patch endpoints when nonlinear input / output curve following is selected via option "C" or "P"; under certain conditions, this produced erratic extrapolations. Values beyond an endpoint are now clamped to the endpoint value, e.g., if lowest gray patch image gray = 18 and this patch corresponds to target reflectance of 0.16, then any sine peak or valley gray values in image < 18 are mapped to 0.16 reflectance in target space.
- Processing flexibility for printer MTF has been expanded. The supplied digital printer target and printer target data file can now be applied to MTF evaluation at any print resolution or scale (previous was only for 500ppi prints); program automatically rescales data in input data file.
- Documentation created with detailed instructions for processing a bar target through *sinemtf*.

version 4.0

web posting: February 2000

- A more robust noise suppression algorithm is implemented by increasing the number of sine rows averaged at a given frequency. [The averaged rows are used to locate sine

peaks and valleys for modulation computation.] Number of rows averaged is a function of the sine frequency, ppi, skew angle, and measurement box size, and is constrained such that the maximum MTF reduction, due to the row averaging, is never greater than 1/2 percent at any given frequency. [V3.1 used frequency-independent row averaging with constant, smaller number of rows averaged.] Compared to V3.1, V4.0 algorithm has a pronounced effect in the low frequency region, where it often smooths-out erratic modulation values; at Nyquist frequency the two versions usually (depending on skew angle) result in the same modulation value. The number of rows averaged is printed-out for each frequency when selecting the "e" (verbose) runtime menu option.

- Vertical and horizontal scales are treated independently. This allows for correct processing of a sine image which was captured with an imaging device that has a real scale difference in the two directions. The two computed directional scale values are used in all computations, rather than the direction-averaged single scale value (latter used in V3.1).

- The density patches in the input data file can now have any labels, e.g., the white patch no longer need be labeled 'F'. However, the whitest image patch MUST correspond to the highest reflectance target patch, otherwise will get runtime message: "Image polarity may be wrong, try re-running with reversed image polarity."
[On the other hand, the blackest image patch need not correspond to the lowest reflectance target patch.]

- The old unix type command line entry has been changed to a menu display with user selection of runtime options. However, for those diehard command-line enthusiasts, there is a 'hidden' command line interface, which works by typing the image properties as arguments on same line as program executable, for example:

```
mtf test.tif tgt.dat 223 236 1223 235 221 728 h
```

Above example will run test.tif image with tgt.dat file, ULcorner at 223, 236; URcorner at 1223, 235; LLcorner at 221, 728; horizontal image orientation.

- The aliasing test is now applied to the frequency range: 30% of Nyquist to Nyquist frequency; V3.1 tested in range: 5 cy/mm to Nyquist frequency.

For given frequency and any skew angle:

C = number of cases to test for aliasing,

C = measurement box height / number of rows averaged,

If C is greater than 10, then C = 10,

then aliasing is said to be present at the given frequency if at least C/3 of the tested cases have detected aliasing,

If $C/3 < 1$ then aliasing is said to be present if just 1 case detects aliasing.

- MTF results are automatically compared to IQS 500 ppi spec or IQS 1000 ppi spec; but if user selects option "n", then output is not compared to IQS spec values.

- For PC/Windows run, the runtime display now appears in a scrollable window.

- Other:

Added output of absolute value of average of vertical and horizontal skew angles.

"Low dynamic range" check deleted (not of practical use).

If "c" or "p" option is selected with "e" option, then the gray level -to-target reflectance values corresponding to the point-to-point piecewise straight line fits are printed for all 256 gray levels.

Code developers are listed in “r” option.

version 3.2

web posting: December 1998

- Changed command line entry to menu display for user selection of options.
- Added “c” option, which forces program to use point-to-point straight line conversion from image gray to target reflectance; useful when device’s input / output response is known to be nonlinear. The “c” and “p” options use the same point-to-point straight line conversion. The “p” option is for printers, whereupon the program asks for the “printer ppi”, which is used in the computation of printer MTF.
- User interface streamlined (less verbose); extraneous verbiage deleted from printout; deleted check for 'low correlation coefficient'.

version 3.1

web posting: December 1996

Base version for public release; complete description can be found in the pdf document file:

MTF_TechRpt.pdf

which is: “Computer Program to Determine the Sine Wave MTF of Imaging Devices”

by N.B.Nill, D.J.Braunegg, and B.R.Paine

MITRE Corporation Technical Report, MTR-96B0000025, June 1996

The MITRE Corporation

202 Burlington Road

Bedford, Massachusetts, USA, 01730

PROGRAM DEVELOPMENT

Software Development: Steve Bennett, Dave Braunegg, Wendy Etkind,
Dave Gonthier, Greg Kearnan, Margaret Lepley,
Norm Nill, Barry Paine, John White, Alex Wu

Concept/Design, Quality Assurance, Project Leader: Norm Nill, Margaret Lepley

Notice

This "sinemtf" software was developed by The MITRE Corporation and was produced for the U.S. Government under Contract numbers J-FBI-12-128, J-FBI-86-028, J-FBI-93-039, DAAB07-99-C-C201, DAAB07-00-C-C201, DAAB07-01-C-C201, DAAB07-01-C-N200, DAAB07-02-C-N200, W15P7T-04-C-D001, and W15P7T-04-C-D199 and is subject to the Rights in Data-General clause FAR 52.227-14, Alt. IV (DEC 2007).

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

-Redistribution of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

-Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.