Instrument Science Report WFC3 2008-014

WFC3 Support in Tiny Tim

Richard Hook & Felix Stoehr (ST-ECF)
June 27, 2008

ABSTRACT

We have extended the Tiny Tim HST point-spread function simulator so that it can now generate PSFs for the Wide Field Camera 3. This ISR describes the changes and sources of information used and shows some comparisons between modeled PSFs and those observed with the CASTLE stimulus during thermal vacuum testing. A web interface to the same simulation software is also available.

Introduction

Tiny Tim [1][3][4] was written by John Krist soon after the Hubble Space Telescope (HST) was launched and has become the most widely used PSF simulator for all the cameras on the observatory. Over the years it has been extended to handle new instruments as they are installed and to incorporate many instrumental effects as they have been discovered. Simulated PSFs are useful for many applications including observation planning, modelling and image deconvolution.

In view of the forthcoming installation of a new camera, the Wide Field Camera 3 (WFC3), during the next servicing mission, later in 2008, it was clearly desirable to have WFC3 included in Tiny Tim before the Phase II proposal period in the summer of 2008. After discussions with the WFC3 team at STScI it was agreed that the ST-ECF would take on this job and build on their experience with the initial NICMOS implementation back in 1997.

This document describes the WFC3 features that have been included in the initial release of Tiny Tim 7.0 and the comparisons that have been made between simulations and point-spread function data from the Thermal Vacuum tests made in Spring 2008. We also introduce a web interface to Tiny Tim.

Characteristics of WFC3

The WFC3 camera will be installed on HST during SM4 in late 2008. It will replace the WFPC2 camera in the axial bay and hence will be close to the optical axis of the telescope. There are two independent optical trains within WFC3, the UVIS channel with two 4096x2048 CCDs with 40mas pixels and the IR channel with a single 1024x1024 HgCdTe detector and 130mas pixels. The optical design is unobstructed apart from a cold-stop within the IR channel. The only refractive elements are windows to the detectors and a weak refractive corrector plate in the IR channel. The internal reflective optics are off-axis designs and both detectors are strongly tilted. This results in significant geometric distortion, comparable to that in ACS, despite the axial location of the camera.

A very detailed description of the instrument and its capabilities can be found in the Instrument Handbook [9].

An early assessment of the optical quality of the UVIS channel, finding it well within the original specification, is described in [8].

Modifications to Tiny Tim for WFC3

In this section we describe the changes we made to Tiny Tim to add WFC3 and describe the details of the information sources used.

Code Modifications

The changes to the Tiny Tim code were relatively minor although extensive. Two new options (22 and 23) are now available in the input menu in *tiny1* for the WFC3 UVIS and IR channels respectively. The use of the software is unchanged and there are no new pure WFC3 features.

In earlier versions of Tiny Tim the only instrument for which geometric distortion was significant was ACS. We have added WFC3 (both channels) to this feature and generalized some of the code in the process. So, when preparing PSFs for WFC3 the Tiny Tim third stage, *tiny3*, which applies geometric distortion and resamples the image onto the detector grid, will normally be required.

Field-dependent Aberrations

We have obtained Zernike coefficients for nine positions on each detector from Zemax modelling from George Hartig (STScI). The values are given with approximate positions in the tables below.

IR detector, RMS waves at 1000 nm

approx. position	(h,h)	(m,h)	(h,m)	(h,1)	(1,h)	(m,1)	(1,m)	(m,m)	(1,1)
X pos	0.0000	0.0135	-0.0122	0.0119	-0.0135	0.0252	-0.0259	0.0014	-0.0015
Y pos	0.0000	0.0135	0.0122	-0.0119	-0.0130	0.0015	-0.0010	0.0259	-0.0250
Z4	0.0048	-0.0152	0.0105	0.0006	-0.0152	-0.0085	-0.0174	-0.0174	-0.0084
Z5	0.0124	0.0075	0.0089	-0.0042	0.0075	-0.0076	0.0026	0.0026	-0.0076
Z6	0.0000	-0.0180	0.0000	0.0000	0.0180	0.0027	0.0138	-0.0138	-0.0027
Z7	-0.0161	-0.0169	0.0060	-0.0118	-0.0048	-0.0224	0.0071	0.0114	0.0135
Z8	0.0167	0.0054	-0.0054	0.0124	0.0175	-0.0129	-0.0109	-0.0065	0.0230

Approximate position: 1=first pixel, h=half of the detector, m=maximum length of the detector
X pos, Y pos: angles in degrees Z4: Focus, Z5: X astigm, Z6: Y astigm, Z7: X coma, Z8: Y coma

UVIS detector, RMS waves at 632.8 nm

position coeff	(h,h)	(m,h)	(h,m)	(h,1)	(1,h)	(m,1)	(1,m)	(m,m)	(1,1)
X pos	0.0000	0.0150	-0.0158	0.0160	-0.0148	0.0311	-0.0304	-0.0010	0.0010
Y pos	0.0000	0.0171	0.0158	-0.0160	-0.0160	0.0011	-0.0010	0.0329	-0.0329
Z4	0.0171	-0.0043	0.0024	-0.0038	0.0016	-0.0183	-0.0002	-0.0277	-0.0284
Z5	0.0000	-0.0066	-0.0178	0.0058	0.0188	-0.0004	0.0020	-0.0252	0.0237
Z6	-0.0064	-0.0106	0.0075	-0.0107	0.0059	-0.0073	0.0367	-0.0085	-0.0099
Z7	0.0000	0.0092	-0.0080	-0.0091	0.0081	0.0011	0.0012	-0.0043	0.0021
Z8	-0.0179	-0.0005	-0.0020	0.0006	-0.0029	0.0297	0.0315	0.0007	0.0005

Positions and coefficients are as above.

After accounting for distortion using the IDCTAB for each detector chip, we fitted a two-dimensional polynomial to the data points using least squares. We treat the UVIS detector as a single virtual chip and assign the points in the gap between the two UVIS chips ((1,h), (h,h), (m,h)) to chip 1. Following the convention in Tiny Tim, we have normalised the positions on the detector to the range 1 to 100 in each direction.

We find the following values and RMS errors for the polynomial coefficients that allow us to compute the aberration *X* at the position (x,y) with

$$Z_X(x,y) = \sum_{i=0}^{n}\sum_{j=0}^{n} Zcoeff_X(i,j)\, x^i\, y^j$$

where n=2 the dimension of the polynomial fit and x,y are pixel-position / detector-side-length in pixels times 100.

IR detector Z4, RMS error=0.003697

I \ j	0	1	2
0	-1.20318149e-02	-1.01187543e-04	7.07476814e-07
1	7.69803251e-04	4.23580589e-08	0.00000000e+00
2	-7.67207830e-06	0.00000000e+00	0.00000000e+00

IR detector Z5, RMS error=0.000521

i \ j	0	1	2
0	-8.35400173e-03	5.13194852e-04	-3.99141208e-06
1	2.01150767e-04	-3.92886164e-08	0.00000000e+00
2	-1.97495797e-06	0.00000000e+00	0.00000000e+00

IR detector Z6, RMS error=0.004789

i \ j	0	1	2
0	1.33063083e-03	1.68392660e-04	-8.55061473e-09
1	-2.58073858e-05	-3.33276120e-06	0.00000000e+00
2	-2.00120320e-08	0.00000000e+00	0.00000000e+00

IR detector Z7, RMS error = 0.001086

i \ j	0	1	2
0	1.25930198e-02	-5.80506052e-04	5.25377957e-06
1	-5.67486735e-04	4.09168135e-06	0.00000000e+00
2	2.15934792e-06	0.00000000e+00	0.00000000e+00

IR detector Z8, RMS error = 0.001086

i \ j	0	1	2
0	2.31206647e-02	1.81284830e-04	-5.30494748e-06
1	-1.39564501e-04	3.97479493e-06	0.00000000e+00
2	-2.06559126e-06	0.00000000e+00	0.00000000e+00

UVIS detector Z4, RMS error=0.001012

i \ j	0	1	2
0	-2.93133070e-02	9.77104130e-04	-7.01112707e-06
1	8.48832287e-04	-3.82000537e-06	0.00000000e+00
2	-7.31339029e-06	0.00000000e+00	0.00000000e+00

UVIS detector Z5, RMS error=0.000175

i \ j	0	1	2
0	2.39527204e-02	3.81747698e-05	-2.58469234e-06
1	-4.91836380e-04	-3.18336492e-07	0.00000000e+00
2	2.48552100e-06	0.00000000e+00	0.00000000e+00

UVIS detector Z6, RMS error=0.001247

i \ j	0	1	2
0	-1.00575676e-02	2.53378515e-04	2.01974800e-06
1	-1.29234209e-04	-4.82961364e-06	0.00000000e+00
2	1.73286690e-06	0.00000000e+00	0.00000000e+00

UVIS detector Z7, RMS error=0.001103

i \ j	0	1	2
0	6.28105522e-04	3.63570927e-04	-3.57989097e-06
1	-3.46734931e-04	-4.54594879e-07	0.00000000e+00
2	3.49171715e-06	0.00000000e+00	0.00000000e+00

UVIS detector Z8, RMS error=0.000830

i \ j	0	1	2
0	1.00672428e-03	-3.97810335e-04	6.97828336e-06
1	-3.53971714e-04	-6.05719636e-06	0.00000000e+00
2	6.57288427e-06	0.00000000e+00	0.00000000e+00

Filters

The filter throughput curves were produced using the synphot files for WFC3 from [5] together with the IRAF task **calcband**. Descriptions of the filters and plots of their throughput curves are given in [9]. The filter curves were normalised to peak of 1.0, trimmed at the edges when the throughput falls below 1% and resampled. We chose to start from the peak pixel and to take every 200th pixel in either direction to match the resolution of the other filter curves in Tiny Tim. For each UVIS filter, the two filter curves that were available - one for each chip - have been merged together using the average value. We have checked that the differences of the throughputs between the chips were negligible. Narrowband filters have been reduced to a single frequency value, again matching the other filter curves in Tiny Tim.

Geometric Distortion

Preliminary estimations of the geometric distortion of the two WFC3 channels have been made from the Zemax optical model of the camera and converted into cubic two-dimensional polynomial coefficients in the standard IDCTAB format used for other HST cameras. These values, kindly made available by Colin Cox (STScI), are included in Tiny Tim 7.0 for use in *tiny3* the last stage in the production of the PSF. They should be regarded as preliminary but are probably good enough for most purposes.

The geometric distortion correction is implemented internally as a conversion from x,y pixel positions to V2,V3 in the telescope focal plane in arcseconds. We have confirmed that this internal code, using the Tiny Tim parameters, agrees with the same conversion done using Python code (based on PyDrizzle) that accesses the IDCTAB directly and is totally independent of the Tiny Tim mechanisms.

For the WFC3 UVIS channel the distortion is a strong skew, similar to that in the ACS. For the IR channel the main effect is a significant difference in the X and Y scales.

Charge Diffusion and Related Effects

CCD detectors suffer from charge-diffusion – photo-electrons created by incident photons are not confined precisely to a single pixel but leak into neighbouring ones. As a result a kernel (that can be well represented as a three-by-three matrix) effectively smoothes the PSF. This kernel is a strong function of wavelength, being a larger effect at shorter wavelengths. For the WFC3 UVIS channel we have included empirical charge diffusion kernels (Table 1) derived from thermal vacuum tests and kindly provided by George Hartig (STScI) [6]. These were derived at 250nm and 810nm and are linearly interpolated between these values. Although the code includes support for space-variant charge diffusion kernels (as implemented for ACS) no information is yet available for WFC3.

0.027	0.111	0.027	0.002	0.037	0.002
0.111	0.432	0.111	0.037	0.844	0.037
0.027	0.111	0.027	0.002	0.037	0.002

Table 1: Charge diffusion kernels for WFC3 UVIS CCD chips at 250nm (left) and 810nm (right).

For the infrared channel there is an Intra-Pixel Capacitance effect that smoothes the PSF and possibly other effects. For this channel we have used a single kernel with no wavelength or space-dependence, also provided by George Hartig. This is shown in Table 2.

0.002	0.038	0.002
0.038	0.840	0.038
0.002	0.038	0.002

Table 2: Diffusion kernel used for the WFC3 IR detector.

The knowledge of the diffusion effects will be refined at a later stage.

Cold-Mask Geometry

The WFC3 IR channel has a cold stop that introduces additional obscuration of the pupil. We have included this effect and modelled it as a circular obscuration of the outside of the pupil, a larger circular central obstruction and wider spiders, aligned with the OTA ones. The cold mask does not cover the pads holding the HST primary. The parameters we used were derived from [7]. The central obscuration radius was 0.354, the outer radius of the cold mask 0.9755 and the spider width 0.061, all expressed in units of the radius of the HST primary at the pupil.

Comparisons between Tiny Tim simulations and Test Data from Thermal Vacuum Tests

During thermal vacuum testing a series of images were acquired using the CASTLE stimulus and a point source through several filters and at several different positions on the detectors of both channels to study the optical quality and PSF characteristics. Some of these data were kindly made available to us by Howard Bushouse and George Hartig (STScI) so that we could perform some simple assessments of the quality of our Tiny Tim simulated PSFs.

The CASTLE imaging train differs in several ways from the actual HST optics. To allow for this we made several changes to the standard Tiny Tim in order to generate simulated PSFs closer to what would be expected from CASTLE:

- The central obstruction size was increased from the nominal HST (0.330) value to 0.455 of the pupil radius.

- The HST mirror maps were turned off.

Although residual aberrations introduced by CASTLE have been mapped using phase retrieval we have NOT included them in our comparisons. This omission is likely to lead to small, but discernable, differences, particularly at short wavelengths.

How the comparisons were made

We ran Tiny Tim with a camera/chip/filter/wavelength/spectrum combination matching the CASTLE data. We used monochromatic filters except for the UVIS PSF of F814W where we used an artificial filter to match the broader bandpass of the light source. We have set the subsampling factor in *tiny3* to 10.

SExtractor [6] was run on the CASTLE image to get the exact, fractional, centre position. We have used the XWIN_IMAGE and YWIN_IMAGE parameters here. Then, the sub-sampled Tiny Tim image was shifted to that position and block averaged back to the original resolution. When producing sub-sampled images, Tiny Tim, of course, does not apply the charge diffusion. We thus convolved the Tiny Tim PSF with the corresponding charge diffusion kernel after the block averaging.

We applied a background subtraction to the CASTLE images using the mean value of all pixels that have either of their coordinates within 10% of the image side length to the border. The CASTLE images are then normalised to the Tiny Tim values using a 1-pixel wide ring of radius two pixels around the centre for matching the intensity scaling. This gave a more robust result than relying on the intensity of, for example, the central pixel. The radial light profiles were computed using the algorithm described in the IRAF task pprofile: The flux in a pixel (regardless of its orientation) is distributed into the two adjacent distance bins using the fractional part of the radius as a weight.

The curves in the radial light profile plots are normalised to a distance of again two pixels. The encircled energy curves are normalised to their last pixel value.
Figures 1-7 show the scaled CASTLE PSFs, the matching Tiny Tim PSFs, the radial intensity profiles (in log-log form) and the encircled energy plots for four UVIS and three IR channel filters.

UVIS Channel Comparisons

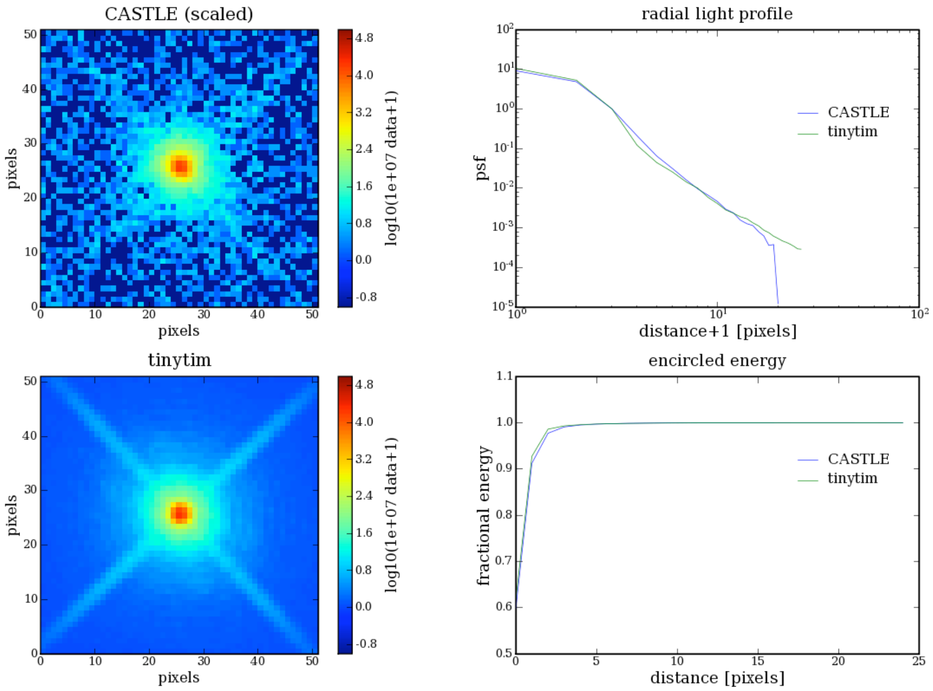


Figure 1: UVIS –F275W

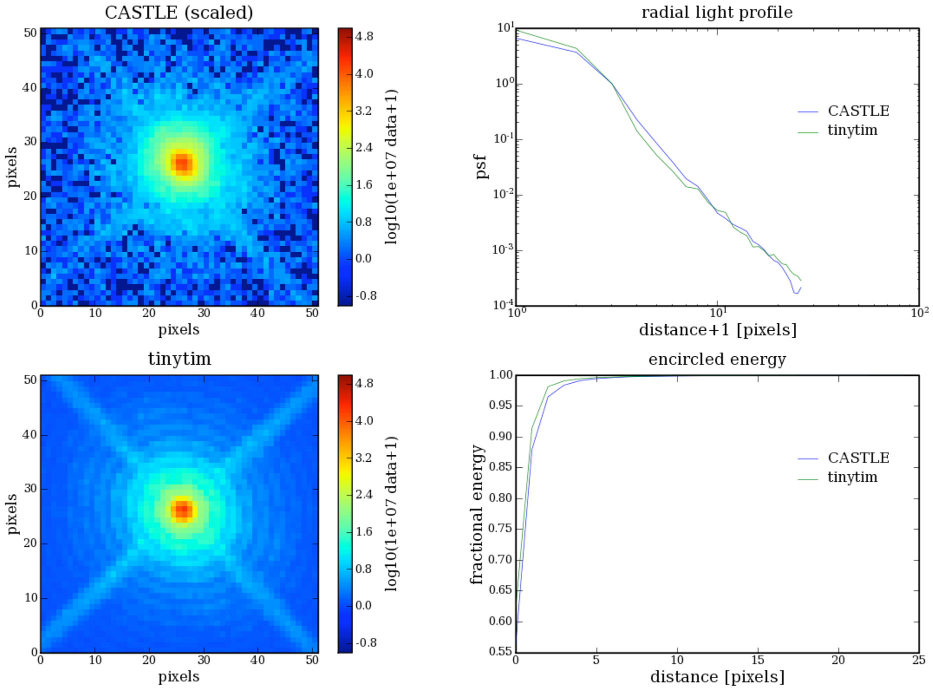


Figure 2: UVIS – F336W

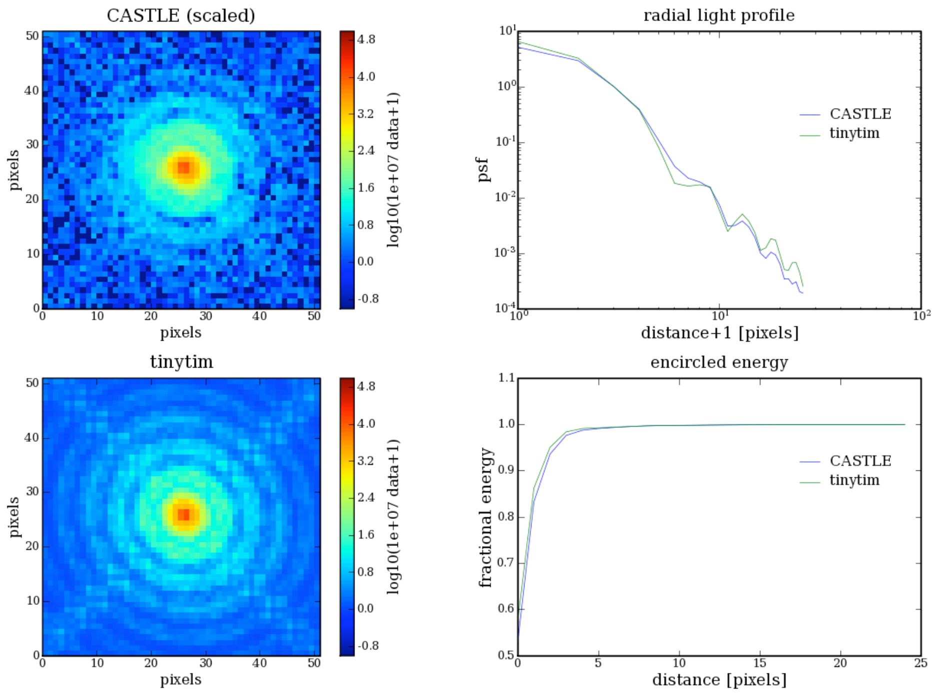


Figure 3: UVIS – F625W

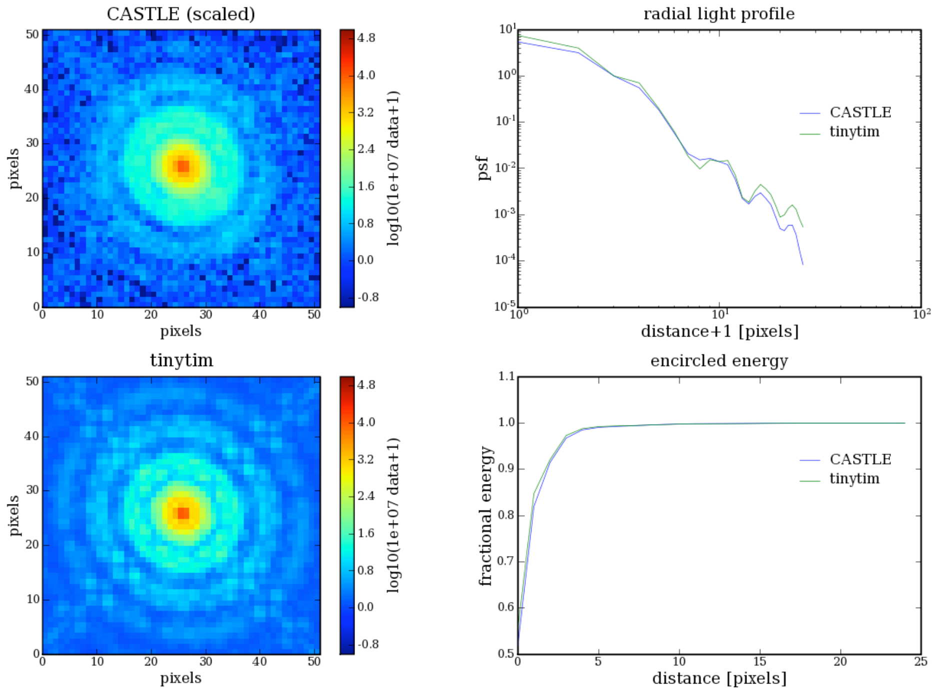


Figure 4: UVIS – F814W

IR Channel Comparisons

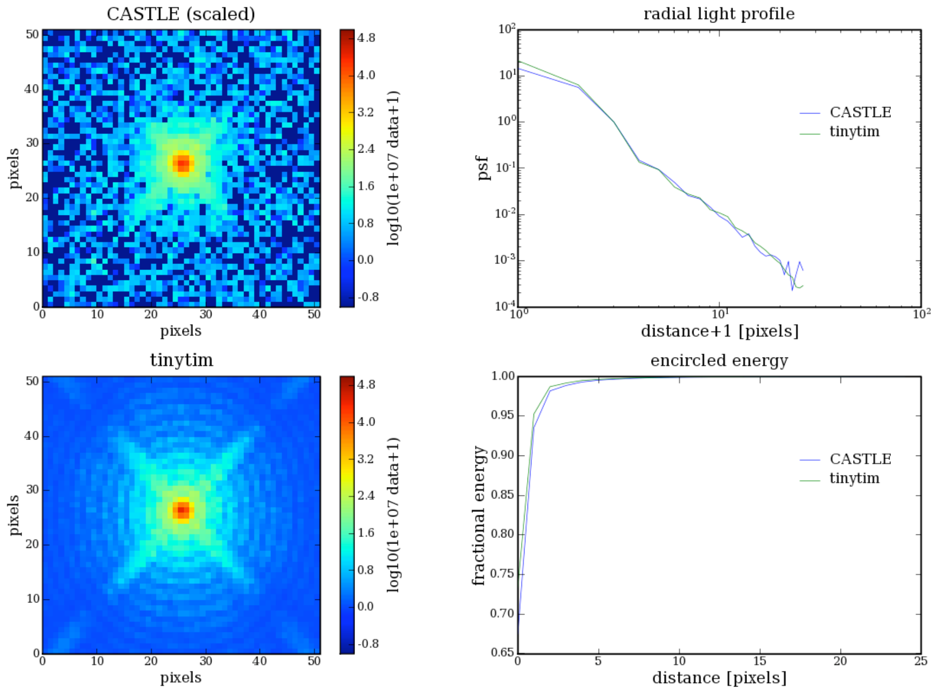


Figure 5: IR - F105W

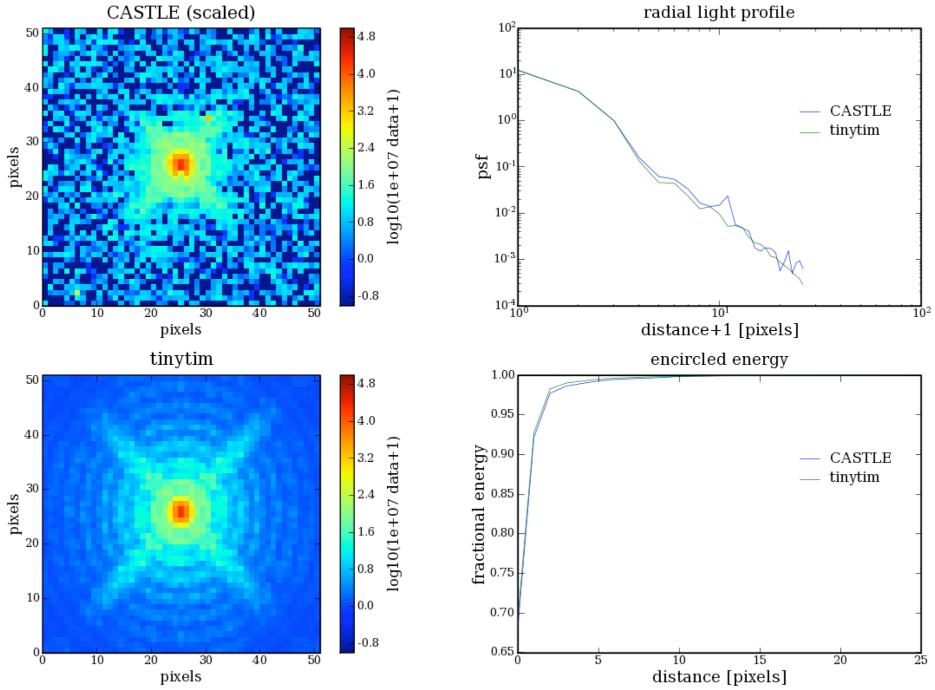


Figure 6: IR - F125W

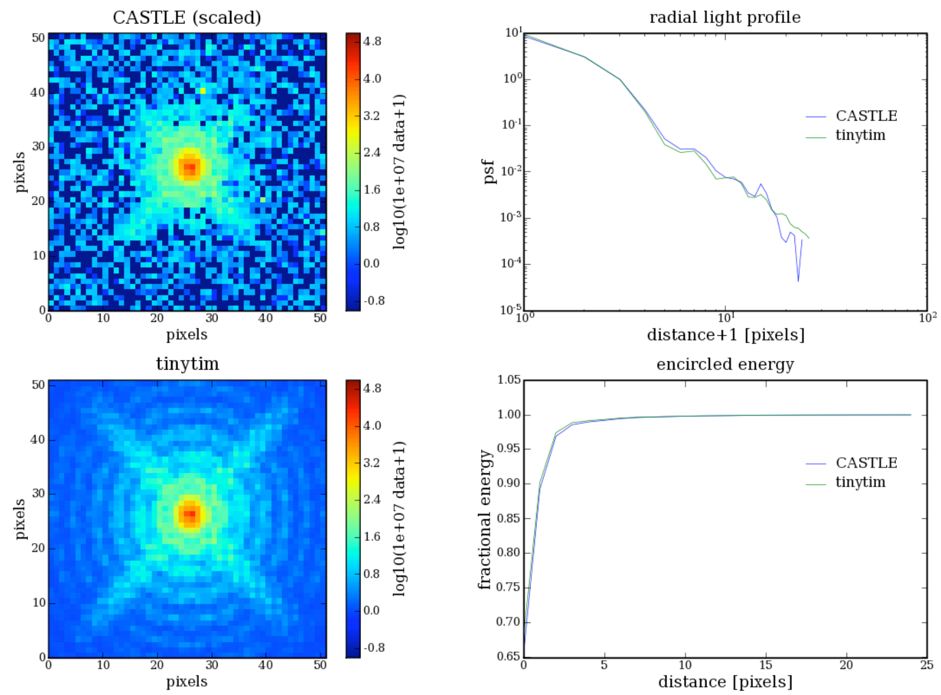


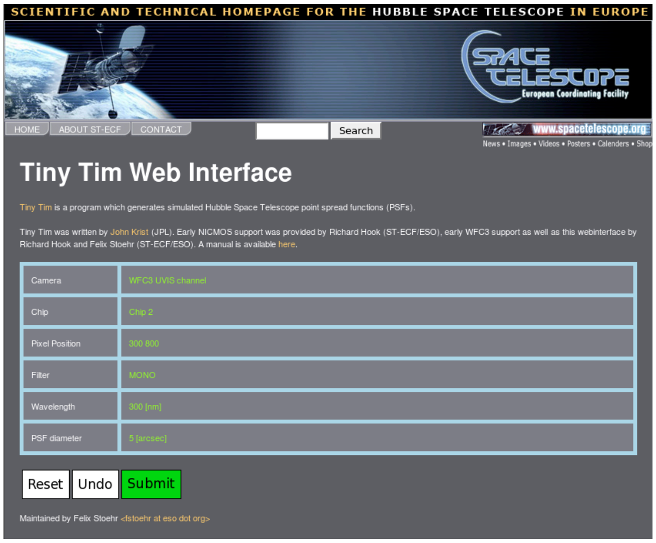
Figure 7: IR – F160W

Obtaining and Installing the WFC3 Version of Tiny Tim

The modified version of the software, the documentation and access to the web version, are all available through the following URL: http://www.stecf.org/instruments/TinyTim/. The installation procedure is identical to that for earlier Tiny Tim versions.

The Web Interface

Although it is easy to download and install Tiny Tim and run it locally, we felt that many would prefer the convenience of a web interface for preparing PSFs. The aim was to have an interface through which the user specifies the PSF parameters using dynamic forms (Figure 8). The PSF is then prepared on the server, using the standard Tiny Tim software and then displays and many options for downloading the resultant image are made available (Figure 9). In addition to the images we also provide data files for the encircled energy and the radial profile as well as all input, output and log files for Tiny Tim. For ACS and WFC3 we provide analysis and images for the distorted and undistorted PSFs. For convenience all files, including the result web pages, can be downloaded for offline browsing in a tar.gz file. The web interface is accessed through the web page at [2].



SCIENTIFIC AND TECHNICAL HOMEPAGE FOR THE HUBBLE SPACE TELESCOPE IN EUROPE

SPACE TELESCOPE
European Coordinating Facility
HOME
ABOUT ST-ECF
CONTACT
Search
www.spacetelescope.org
News • Images • Videos • Posters • Calenders • Shop

Tiny Tim Web Interface

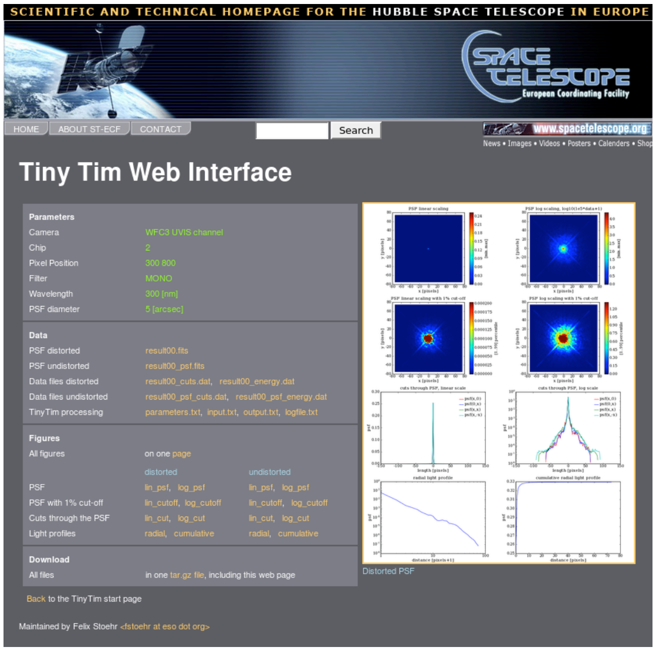
Tiny Tim is a program which generates simulated Hubble Space Telescope point spread functions (PSFs).

Tiny Tim was written by John Krist (JPL). Early NICMOS support was provided by Richard Hook (ST-ECF/ESO), early WFC3 support as well as this webinterface by Richard Hook and Felix Stoehr (ST-ECF/ESO). A manual is available here.

Camera	WFC3 UVIS channel
Chip	Chip 2
Pixel Position	300 800
Filter	MONO
Wavelength	300 [nm]
PSF diameter	5 [arcsec]

Maintained by Felix Stoehr <fstoehr at eso dot org>

Figure 8: Web form itself, after it is complete and ready to run.



SCIENTIFIC AND TECHNICAL HOMEPAGE FOR THE HUBBLE SPACE TELESCOPE IN EUROPE

SPACE TELESCOPE
European Coordinating Facility
HOME
ABOUT ST-ECF
CONTACT
Search
www.spacetelescope.org
News • Images • Videos • Posters • Calenders • Shop

Tiny Tim Web Interface

Parameters		
Camera	WFC3 UVIS channel	
Chip	2	
Pixel Position	300 800	
Filter	MONO	
Wavelength	300 [nm]	
PSF diameter	5 [arcsec]	
Data		
PSF distorted	result00.fits	
PSF undistorted	result00_psf.fits	
Data files distorted	result00_cuts.dat, result00_energy.dat	
Data files undistorted	result00_psf_cuts.dat, result00_psf_energy.dat	
TinyTim processing	parameters.txt, input.txt, output.txt, logfile.txt	
Figures		
All figures	on one page	
	distorted	undistorted
PSF	lin_psf, log_psf	lin_psf, log_psf
PSF with 1% cut-off	lin_cutoff, log_cutoff	lin_cutoff, log_cutoff
Cuts through the PSF	lin_cut, log_cut	lin_cut, log_cut
Light profiles	radial, cumulative	radial, cumulative
Download		
All files	in one tar.gz file, including this web page	

Back to the TinyTim start page

Distorted PSF

Maintained by Felix Stoehr <fstoehr at eso dot org>

Figure 9: The final display of the Tiny Tim web interface after a PSF has been created on the server. The left side lists a selection of products to be downloaded and the right shows several visualisations of the generated PSF.

Users of the web interface are strongly advised to carefully read the relevant parts of the User Manual [3] before preparing any PSFs.

Future plans

Tiny Tim 7.0, with the initial WFC3 support, was made available in early June 2008 from the usual STScI Tiny Tim web site [1] as well as through the ST-ECF web pages [2]. It must be stressed that this version is based on early estimates for aberrations, geometrical distortions and charge diffusion effects and similar effects in both channels and is certainly far from complete. As a result the PSFs created may differ significantly from those found once the camera is in orbit. We intend to include better estimates as they become available and also try to incorporate the "unknown unknowns" that surface as the camera is used for science.

References

[1] The Tiny Tim site at STScI: http://www.stsci.edu/software/tinytim/tinytim.html

[2] Web access at the ST-ECF: http://www.stecf.org/instruments/TinyTim/tinytimweb

[3] The Tiny Tim Users Guide: http://www.stsci.edu/software/tinytim/tinytim.pdf

[4] Krist, J. 1995, "Simulation of HST PSFs using Tiny Tim", Astronomical Data Analysis Software and Systems IV, ASP Conference Series, R.A. Shaw, H.E. Payne, and J.J.E. Hayes, eds., 77, 349.

[5] WFC3 synphot files: ftp://ftp.stsci.edu/cdbs/comp/wfc3/

[6] Bertin, E. and Arnouts, S. 1996, SExtractor: Software for Source Extraction, Astronomy and Astrophysics Supplement, 117, 393

[7] Randal Telfer 2006, "A Nominal Geometrical Calculation of the Throughput of the WFC3 IR Channel Cold Mask", unpublished internal memo, kindly supplied by George Hartig.

[8] Hartig, G. and Baggett S., 2004, STScI WFC3 ISR 2004-08

[9] Bond, H.E., and Kim Quijano, J., et al. 2007, "Wide Field Camera 3 Instrument Handbook, Version 1.0" (Baltimore: STScI)