2003

Deconvolving Images of Titan

Maria Alikakos Fout

COLLEGE SCHOLARS PROJECT APPROVAL

Maria Alikakos Fout
Scholar

Dr. William Blass
Mentor

Deconvolving Images of Titan – April 27 2003
Project Title and Completion Date

COMMITTEE MEMBERS' SIGNATURES
(Minimum 3 Required)

W. F. Blass
W. F. BLASS

Marianne Breinig

[signature]

PLEASE ATTACH A COPY OF THE SENIOR PROJECT TO THIS SHEET AND RETURN BOTH TO THE PROGRAM DIRECTOR. THIS PAGE SHOULD BE DATED AND COMPLETED ON THE DATE THAT YOUR DEFENSE IS HELD.

DATE COMPLETED May 3, 2003

Deconvolving Images of Titan

College Scholars Project

Maria Alikakos Fout

May 2003

Table of Contents

1. Introduction……3
1.1 theory……3
1.2 role of guide star……4

2. Methods……6
2.1 recording the images……6
2.2 choosing the wavelengths……7
2.3 implementation……8

3. Results……9
3.1 titan……9
3.2 images……9
3.3 analysis……14

4. Discussion……15
4.1 application……15
4.2 cassini-huygens……15

Appendix……17
A.1 programs……18
A.2 sketches……23
A.3 program parameters……26

ABSTRACT

Observation of celestial objects from the earth's surface, regardless of the power of the instrument used for observation, yields images that exhibit distortion. This distortion is a result of two critical factors. The first of these, which results from the obscuring effect of the earth's atmosphere, can be represented by a "blurring" function. The second factor results from the effect of the optical system used, which causes a certain degree of degradation of the stimuli received. This degradation is described by the point-spread function, which is unique to each instrument. In the end, the registered data is the product of the light from the object of interest convolved with the blurring function of the atmosphere and the point spread function of the instrument employed. This paper describes some aspects of the CBM post-observation adaptive imaging process created by W.E. Blass and Stephen L. Mahan of the University of Tennessee, and Gordon Chin of the Goddard Space Flight Center, which corrects for the effects of blurring and point spread functions on an image. In addition, a description of the application of the CBM method on images of Titan is included, as well as an analysis of the results.

1. Introduction

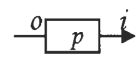
1.1 theory

Although scientific instruments are capable of only discrete sampling techniques, the optical images these instruments attempt to record are often modeled as continuous functions of parameters such as time or frequency. The observable image can be modeled by a continuous integral:

$$i(x, y) = \iint p(x' - x, y' - y)\; o(x', y')\, dx'\, dy' \tag{1}$$

or as a linear system

$$i = p \otimes o \tag{2}$$

where $\otimes$ is the convolution operator, o is the object being imaged (system input), i is the image (system output), and p is the point spread function (psf) which represents a linear system model  .

It is assumed that $p \approx p_{instr} \otimes p_{atm}$ where p_{instr} and p_{atm} are the instrument's point spread function and the atmosphere's blurring function respectively.

The instrument's point spread function can be modeled or measured. Ideally the inverse point spread function p^{-1} would be found from

$$p \otimes p^{-1} = \delta \tag{3}$$

where δ is the Dirac delta function. In Fourier Transform space this is

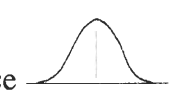
$$\hat{p} * \hat{p}^{-1} = \hat{\delta} \tag{4}$$

and

$$\hat{p}^{-1} = \frac{\hat{\delta}}{\hat{p}} \tag{5}$$

so that we may use the p^{-1} in convolution with the blurred image to solve for the unblurred object o:

$$p^{-1} \otimes i = o \tag{6}$$

However equation (6) presents a problem since the extension of p in Fourier transform space  is finite, while that of δ  is infinite. To circumvent this problem, a target function $\mathcal{T}$ is chosen, generally a very narrow Gaussian function, because this yields resolution enhancement independent of the data. When this $\mathcal{T}$ is convolved with equation (3), we have:

$$\underline{\mathcal{T}\otimes p^{-1}}\otimes p=\delta\otimes\mathcal{T}=\mathcal{T} \qquad (7)$$

Then defining $\mathcal{T}\otimes p^{-1}$ as q, *(7)* becomes

$$q\otimes p=\mathcal{T} \qquad (8)$$

After q has been solved for

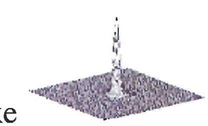
$$\begin{aligned} q\otimes i &= q\otimes o\otimes p \qquad \text{(from}(1)) \qquad (9) \\ &= \mathcal{T}\otimes p^{-1}\otimes o\otimes p \quad \text{(from}(7)) \\ &= \mathcal{T}\otimes p^{-1}\otimes p\otimes o \end{aligned}$$

and finally

$$q\otimes i = \mathcal{T}\otimes o \qquad (10)$$

In words, an image i of the object o processed with p can be convolved with q to give the same result that would be obtained with an instrument having psf = $\mathcal{T}$ recording o. Essentially q is an estimator of the inverse of p, and as such is used as a linear filter. This way, to retrieve the deconvolved data, q is convolved with the observed data. Using this method, only knowledge of p is required to obtain q. Once q is obtained, it can then be applied to all data observed with that same instrument within a small time-frame (so as to approach the same atmospheric conditions).

1.2 role of guide star

As mentioned earlier the factors affecting the image of a celestial object are blurring, caused by the atmosphere, and the optical psf of the telescope. Consider the case of a single resolvable moderately bright star, which will be called the "guide star". Due to the magnitude of the distance from us, it can be assumed that, in the absence of any distortion, the image of the star would consist of a single point source of light. In three dimensions this point source is visualized as a δ-function spike . In practice however the expected δ-function spike is instead replaced by what more closely resembles a Gaussian shaped mountain  (or worse).

The overall effective psf is some combination of p_{instr} and p_{atm}. These are assumed to comprise the agents that flatten the expected star's δ_{star}-function spike to the image "mountain" i_{star} that we get instead. Therefore, while a hypothetical perfect instrument looking through an atmosphere with no turbulence would give us the true image of the object of interest, δ_{star}:

$$p_{ideal} \otimes o_{star} = \delta_{star} \qquad \text{(where } o=\delta\text{, for the reasons described above)} \qquad (11)$$

instead, we get

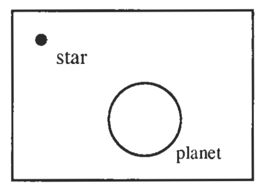
$$p \otimes o = i_{star} \qquad \text{(where } i_{star} \text{ resembles the aforementioned mountain)} \qquad (12)$$

So then combining equations (11) and (12)

$$p \otimes \delta = i_{star} \qquad (13)$$

Now we want to solve for p^{-1} so that $i_{star} \otimes p^{-1} = \delta_{star}$ by using the mathematical method described in section 1.1. If this p^{-1} "corrects" i_{star} to the expected δ_{star}, then any other object in the same field as the star should also be corrected with the same p^{-1}.

So if the image field is as shown in the figure to the right, by finding p^{-1} to correct i_{star} to δ_{star}, then p^{1} should also deblur i_{planet}, since it is also in the same field as the star. Thus the blurred image of a planet can be deconvolved by finding the filter which transforms the guide star into a δ-function convolved with $\mathcal{T}$, and then passing the image of the planet through that filter. This is the method that was used to deconvolve the images of Titan presented later on in this paper.



2. Methods

2.1 recording the images

The images of Titan referred to in this paper were recorded using an acousto-optic tunable filter camera coupled with the Mt. Wilson 100-inch Hooker adaptive optics telescope. A simplified diagram of this system is shown in Figure 1 below.

Adaptive Optics Telescope

The light emitted from a celestial object reaches the telescope as a procession of wavefronts, which are deformed by the turbulence in the earth's atmosphere. The adaptive optics telescope partly makes up for this deformation in real time by measuring the atmospheric distortions and sending electronic signals to a deformable mirror, which can change its shape rapidly to correct for the distortions.

AOTF

An acousto-optic tunable filter's main component is a transparent birefringent crystal excited by a radio frequency transducer. Propagating acoustic waves inside the crystal create regular spatial variations of the refractive index. Light of a particular wavelength incident on the crystal is diffracted by the moving grating produced by the acoustic wave.

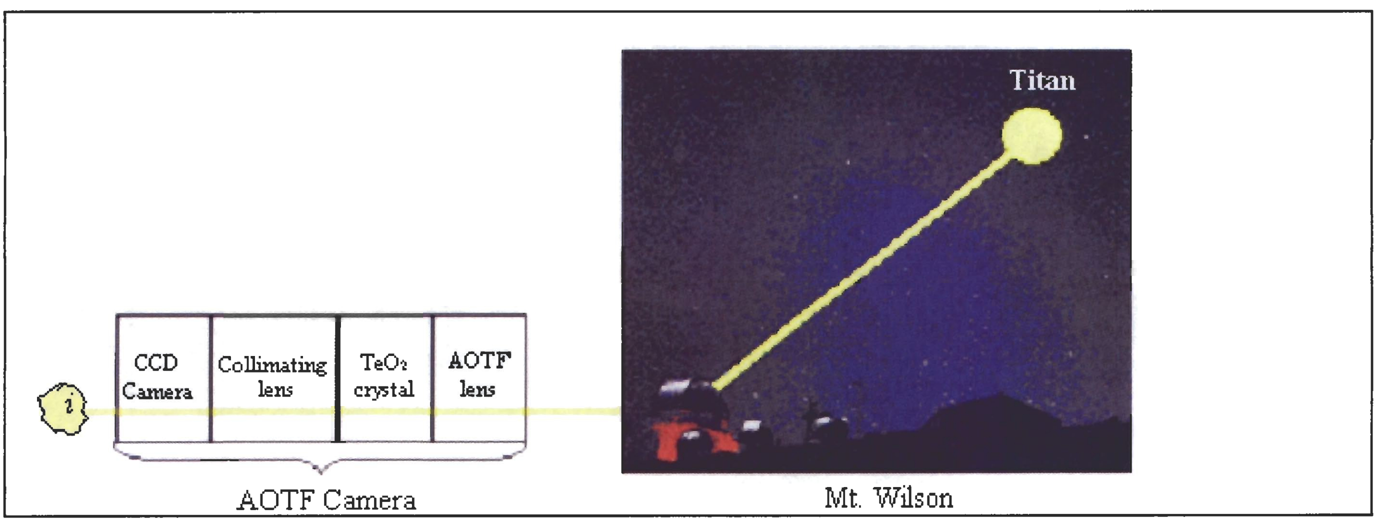


Figure 1 Light emitted from Titan reaches the Mt. Wilson telescope and passes through the AOTF camera before being recorded

Using an AOTF, both (i) the amount of light diffracted and (ii) the spectral frequency diffracted can be selected. By making the AOTF part of an imaging system and projecting the diffracted light onto a 2-D array (CCD), an image is formed.

The wavelengths selected for the observations of Titan were 830, 890, 920, 935, 930, 935, 940, 945, 950, 960 and 1030 nm.

CCD Camera

A charge coupled device converts an incoming photon of light to an electron, which is then stored in the detector's array of pixels until it is read out in the form of a spreadsheet with groups of numbers in each cell representing the number of electrons gathered in each pixel. Different pixels on the CCD gather different amounts of electrons (photons) from the incoming image of the object under observation. The range of numbers describing these amounts at each pixel of the CCD is displayed by computers as shades of gray for each pixel site on the screen, thus producing the image one sees.

2.2 choosing the wavelengths

Titan has a thick atmosphere (1.5 bar) dominated by molecular nitrogen in which an active photochemistry occurs producing hydrocarbons, nitriles and more complicated organic gases, including oxygenated compounds. Ultraviolet light transforms these gases into a thick, smog-like haze that lays in the upper atmosphere, making the stratosphere impenetrable to light in the visible spectral range. However, infrared light can penetrate this smog. Specifically, certain infrared wavelengths serve as transparent "windows" through the methane haze (see Figure 2). In the window ranges, the combination of low gas opacity and low haze absorption allows one to probe the lower atmosphere and surface of Titan to get some sense of what the actual surface might be like. Alternately, the complete range of methane absorption across the wavelength region 600nm – 1100nm allows the spectra to probe all levels in the atmosphere.

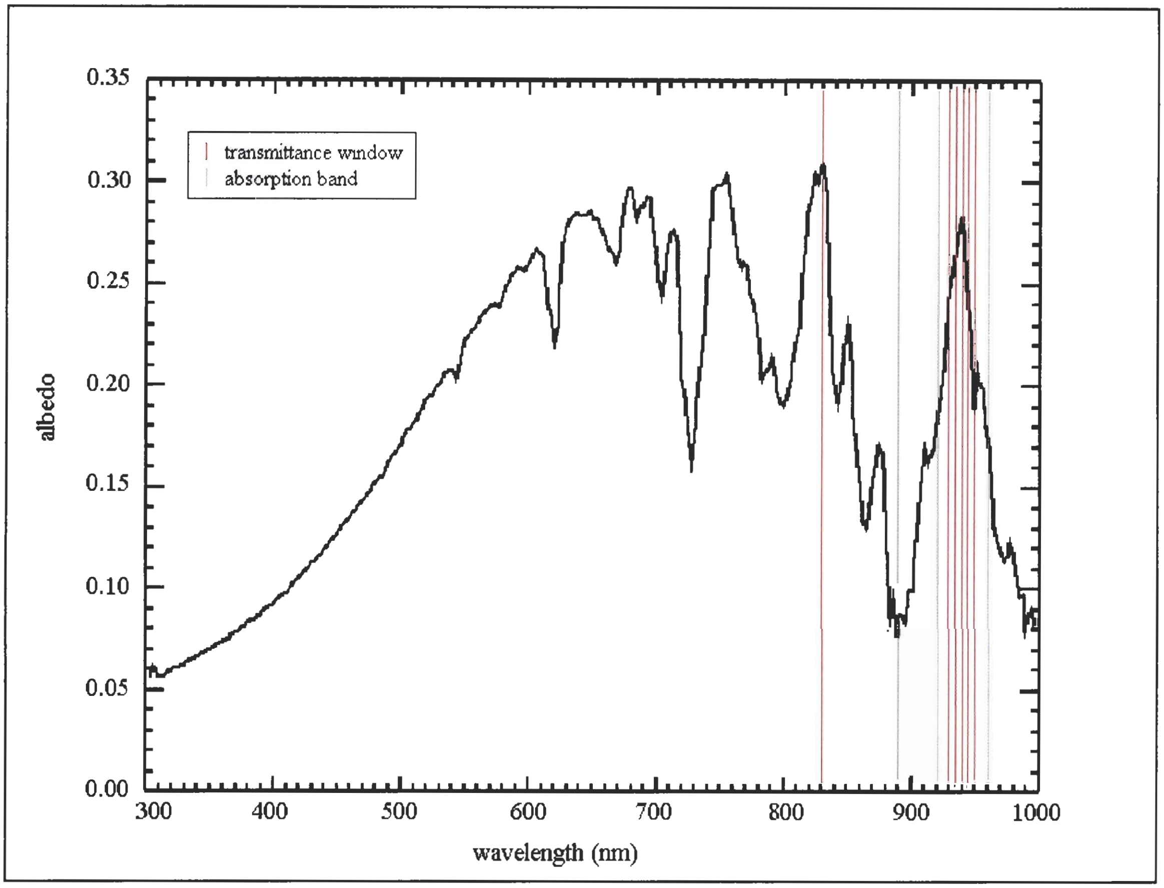


Figure 2 The graph above displays (i) the wavelengths at which the methane bands exist, where there is stronger absorption so that no light returns from below the 100 km altitude, and (ii) the wavelengths at which the methane "windows" exist, allowing light to probe the lower atmosphere and surface. The images presented in this paper capture both the haze of Titan at different altitudes (gray) and its lower atmosphere/surface (red).

2.3 implementation

The main enhancement program was written in IDL (Interactive Data Language), software designed for data analysis, visualization, and cross-platform application development. The input to the program is (i) the image to be enhanced and (ii) the user-defined full width at half maximum of the target function $\mathcal{T}$ (Gaussian). The output is (i) the enhanced image and (ii) the point-spread function. The process of enhancing the image of Titan involves two passes of the program. In the first pass the program is run with the image of the guide star. Values of the full width at half maximum (fwhm) are iterated until a satisfactory transformation is achieved. The parameters of this transformation are then used in a second pass to transform the image of Titan. The grayscale images produced by the program are finally passed through a transfer function designed to highlight variations in intensity.

The source code of the programs is included in the Appendix.

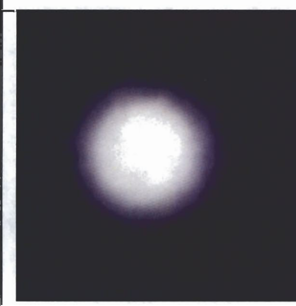
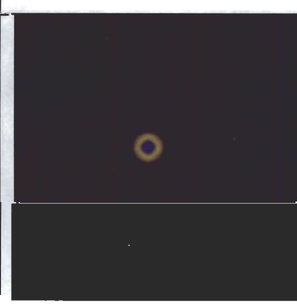
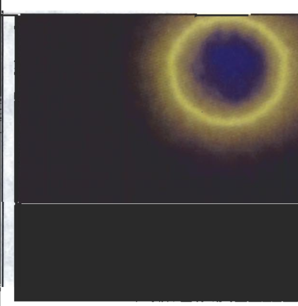
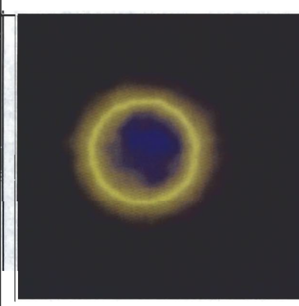
3. Results

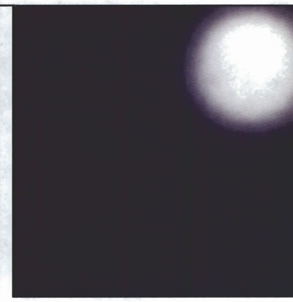
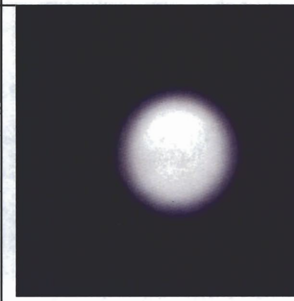
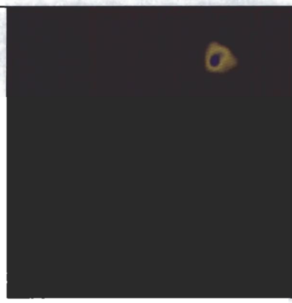
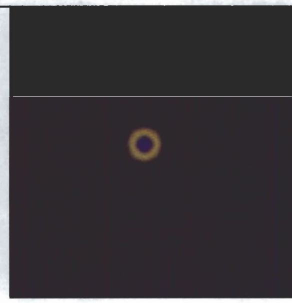
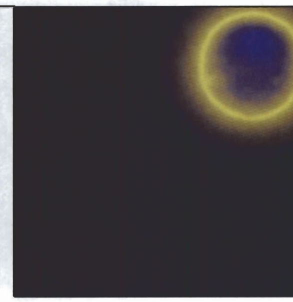
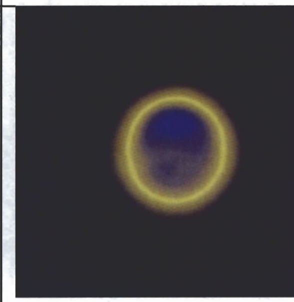
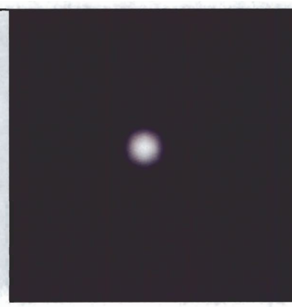
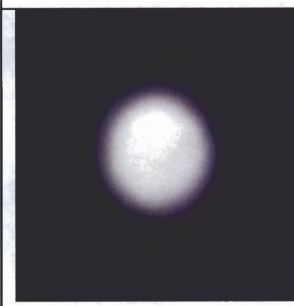
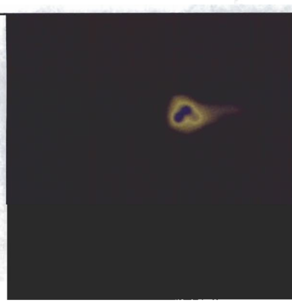
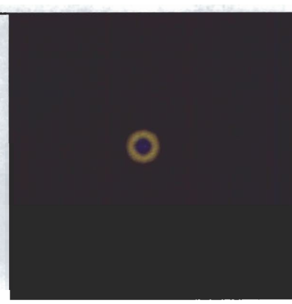
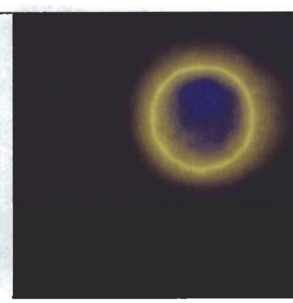
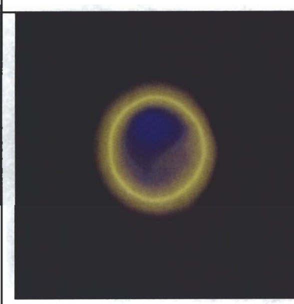
3.1 titan

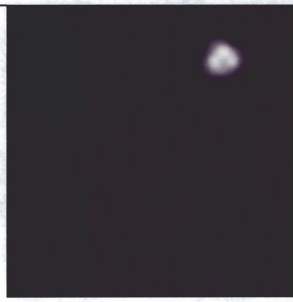
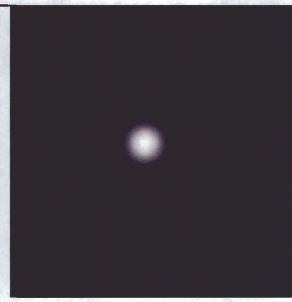
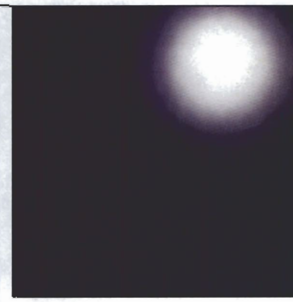
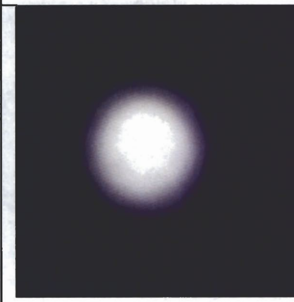
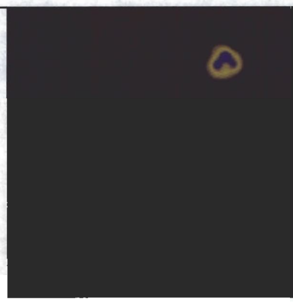
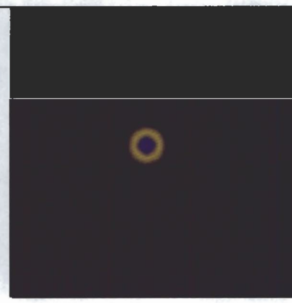
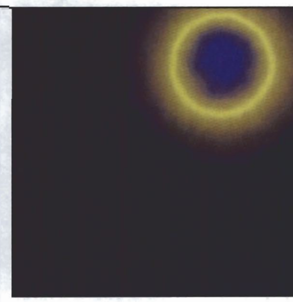
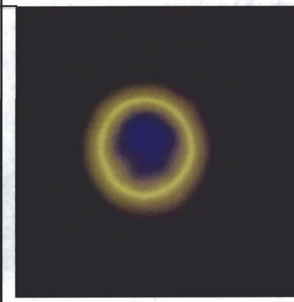
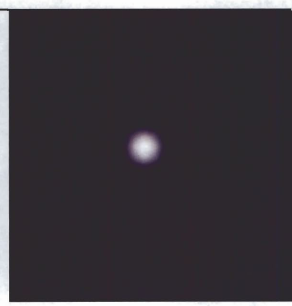
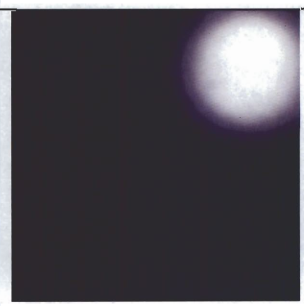
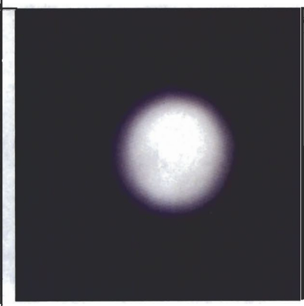
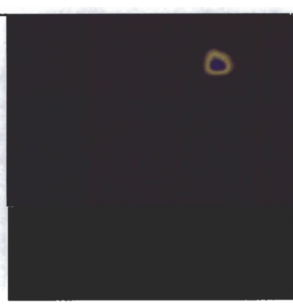
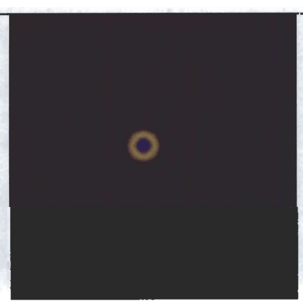
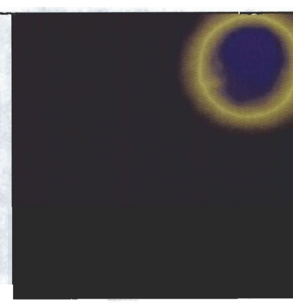
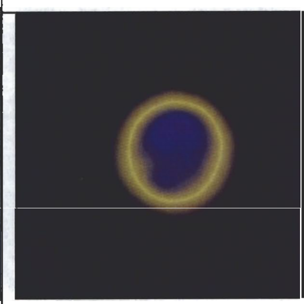
Titan is the largest of Saturn's thirty moons, and is the fifteenth closest to the planet. Titan happens to be the only satellite in our Solar system that has a thick and extended atmosphere, with nitrogen as its main gaseous constituent, along with small amounts of methane and a little molecular hydrogen. Within the Solar System Titan resembles our planet most closely and it is the best possible natural laboratory to study conditions on the primordial Earth.

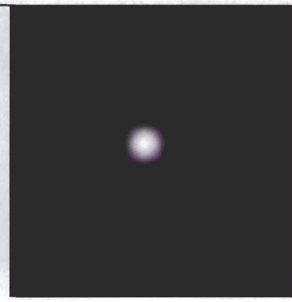
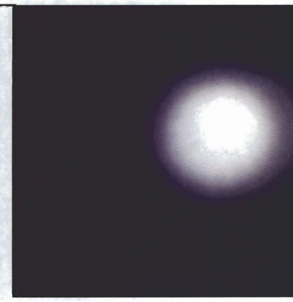
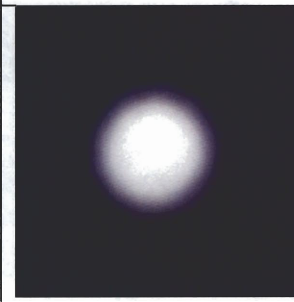
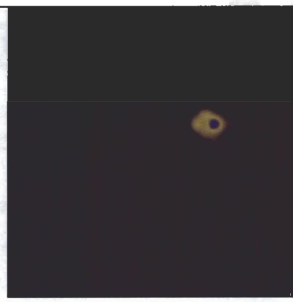
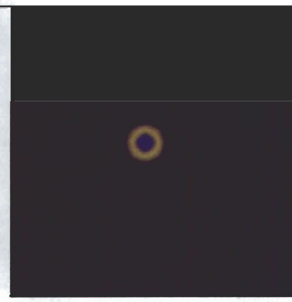
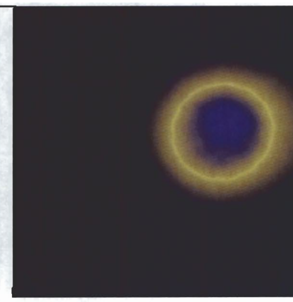
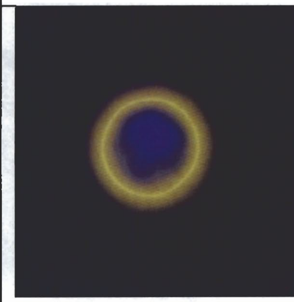
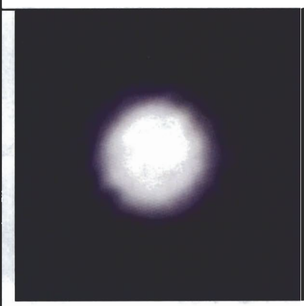
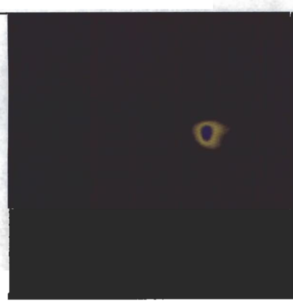
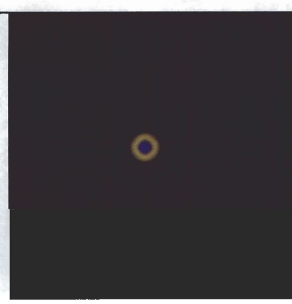
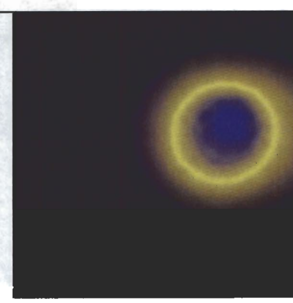
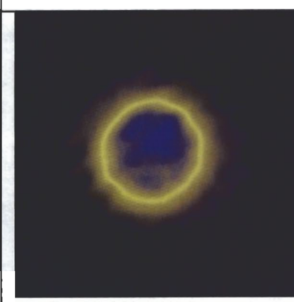
3.2 images

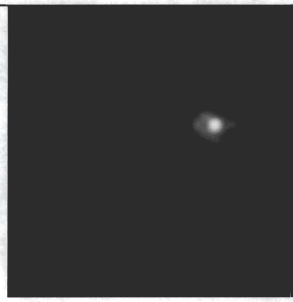
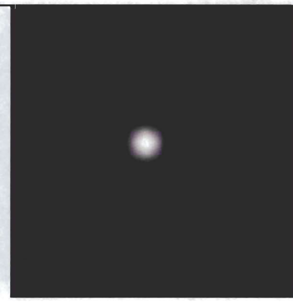
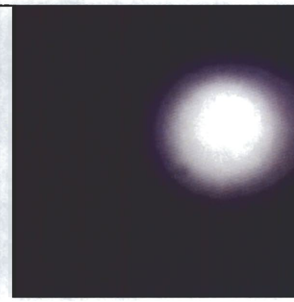
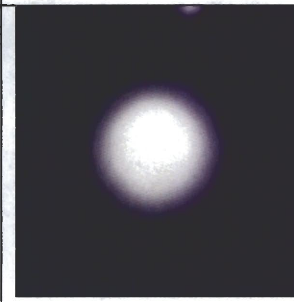
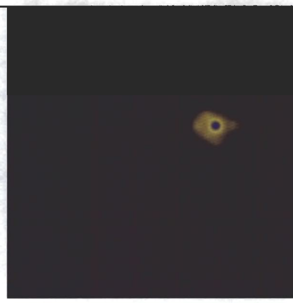
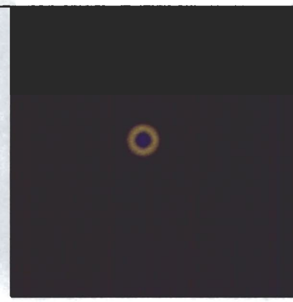
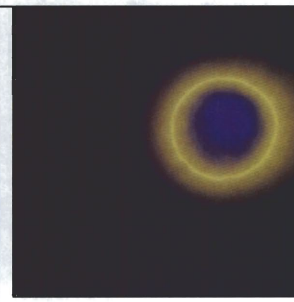
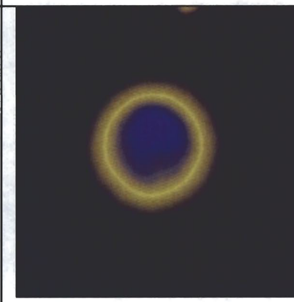
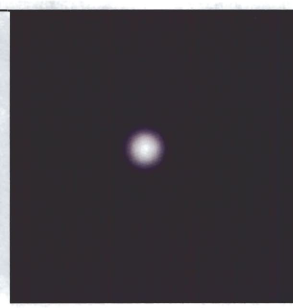
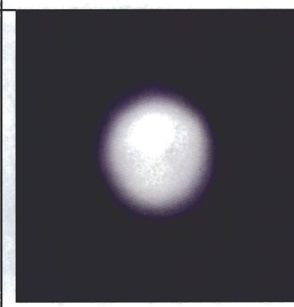
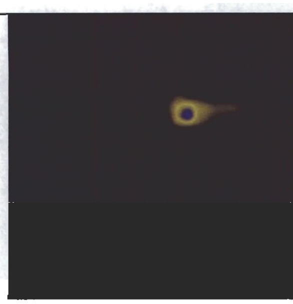
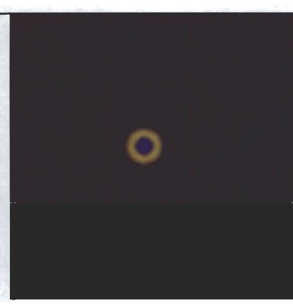
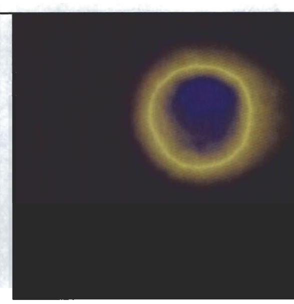
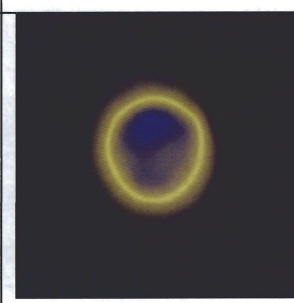
The guide star whose psf was used is H718, and is depicted in the two left columns in the table below. The images of Titan are displayed in the right two columns. Each row of gray-scale images is followed by a row of RGB images.

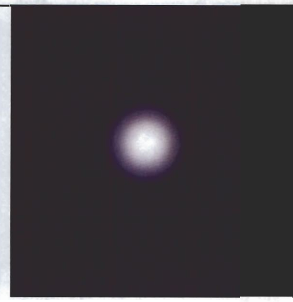
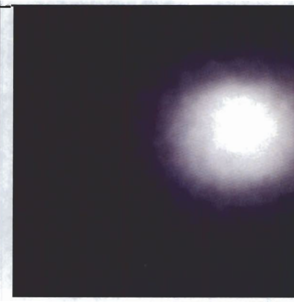
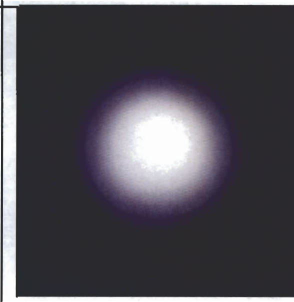
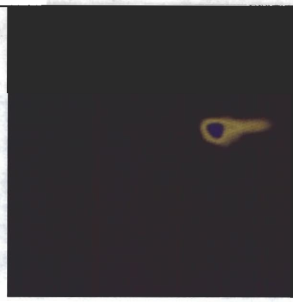
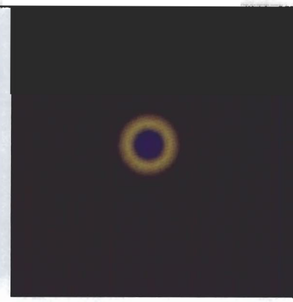
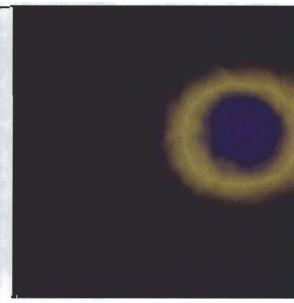
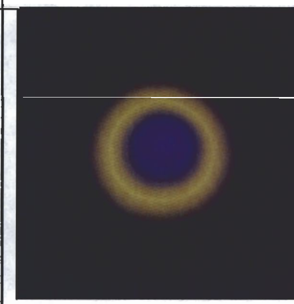
	H718 Original Image	**H718 Recovered Image**	**Titan Original Image**	**Titan Recovered Image**
absorption bands				
890 nm				
				
				

	H718 Original Image	H718 Recovered Image	Titan Original Image	Titan Recovered Image
absorption bands				
920 nm				
				
				
960 nm				
				
				

	H718 Original Image	H718 Recovered Image	Titan Original Image	Titan Recovered Image
transmittance windows				
830 nm				
				
				
930 nm				
				
				

	H718 Original Image	H718 Recovered Image	Titan Original Image	Titan Recovered Image
transmittance windows				
935 nm				
				
				
940 nm				
				
				

	H718 Original Image	**H718 Recovered Image**	**Titan Original Image**	**Titan Recovered Image**
transmittance windows				
945 nm				
				
				
950 nm				
				
				

	H718 Original Image	H718 Recovered Image	Titan Original Image	Titan Recovered Image
transmittance windows				
1030 nm				
				
				

3.3 analysis

Although there seems to be a wavelength dependence of contrast, certain "dark" and "light" regions on Titan seem to be constant throughout the majority of the images. Since methane on Titan is the counterpart of water on earth, it is plausible that these regions represent ice-mountains and valleys on the moon. Furthermore, assuming a meteorological system similar to Earth's, it is plausible that the methane clouds rain on the moon, forming lakes and oceans, which might explain the brightest regions in the images above.

The effects of noise can be seen clearly in the dark/sky regions surrounding the objects, from which it can be concluded that the generation of extraneous features on the object by noise is unlikely.

4. Discussion

4.1 applications

The method discussed in this paper provides a low-cost means to achieving high quality image enhancement and can be applied to all types of instruments, regardless of power. Also, it is a post-imaging process, making it truly instrument independent. Furthermore, the theory on which this method is based can be applied to other modalities such as NMR for chemical spectra or MRI for medical data.

4.2 cassini-huygens

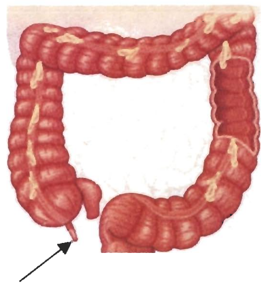
The spacecraft Cassini was launched in October of 1997, and is scheduled to reach Saturn's orbit in June of 2004, where it will be able to gather information about the planet's rings, its magnetosphere, and its moons, Titan in particular. In November 2004, Cassini will drop a probe called Huygens into Titan's upper atmosphere. As Huygens breaks through the cloud deck, a camera will capture pictures of the Titan panorama. Other instruments will measure the chemical composition of Titan's atmosphere as the probe descends to the surface. Huygens is designed to bounce and float, so it is prepared for whatever surface it finds on Titan. However, Huygens will send information for only a few hours, as it must operate on batteries (Titan's haze is too thick to allow operation under solar power). Cassini is scheduled to remain in orbit around Saturn and Titan for years.

The Cassini-Huygens mission will shed light on the accuracy of the methods implemented in this paper. If the results in this paper are in agreement with the mission's findings, then these methods can be applied with confidence to more distant celestial objects of interest.

5. Acknowledgements

I would like to thank Dr. Blass for giving me the opportunity to work on this project, and the chance to experience, as an undergraduate, what it is like to be in a research position.

Appendix



A.1

IDL program to enhance images

```
pro maria2
;uses congrid to expand (128,128) to (256,256)
close,/all
indim=256
dim=256
data=fltarr(256,256)

;read in psf 256x256, and normalize it
psf=fltarr(dim,dim)
print,'Select Instrument PSF'
psfname=pickfile()
psf=readfits(psfname)                          ;reads in psf file (p)

;psf=psf>0.
psf=psf/total(psf)

;read image to enhance
print,'select image to enhance'
fname=pickfile()
data = readfits(fname)                         ;reads in file for enhancement. (i)

;data=data>0.

print,'psf file= ',psfname
print,'image= ',fname

;choose width of target Gaussian (T
print,'Input-max-target-FWHM, decrement {2 numbers}'
read,fwhm,decrement

loop:
fwhm=fwhm-decrement

fw=fwhm*10
fw=string(fix(fw))
fw=strtrim(fw,2)

print,'Mapping will be created for Targets having FWHM of:', fwhm

;enhance image using algorithm and fwhm
x=shift(dist(dim),dim/2-1,dim/2-1)
tar=exp(-x^2/(2.*(fwhm/2.35)^2))
tar=tar/total(tar)
ftar=fft(tar,-1)
fpsf=fft(psf,-1)
fq=ftar/fpsf
window,0,xs=512,ys=512,title='Original'           ;display original
window,1,xs=512,ys=512,title='Recovery '          ;display enhanced
        wset,0
tvscl,data
brec=gauss2dfit(data,bout,/tilt)
print,'orig',bout(2),bout(3),bout(6)
fdata=fft(data,-1)

;recover image data and get array ready for display
rec=abs(fft(fdata*fq,1))
arec=gauss2dfit(rec,aout,/tilt)
```

```
print,'recovery',aout(2),aout(3),aout(6)


;check the volume (Integrated Intensity) of the images
 print,'Recovery  Integrated Intensity=', $
              total(rec)
 print,'Original Integrated Intensity=', $
              total(data)
basename=strtrim(fname)
print,'Dataset #',basename+fw,' Completed'

;display the data and recoveries
parts=str_sep(basename,'.',/remove_all)
              wset,0
tdata=congrid(data,2048,2048,cubic=-.5)>0.
tvscl,tdata(1000:1512,768:1362)>0
write_tiff,parts(0)+'orig.tif',tvrd()
              wset,1
trec=congrid(rec,2048,2048,cubic=-.5)>0.

tmp=trec(768:1280,768:1280)>0
tvscl,tmp>0
xyouts,10.,10.,fwhm,/device
write_tiff,parts(0)+'recov'+fw+'.tif',tvrd()
writefits,parts(0)+'-'+fw+'-rec.fits'
window,2
show3,tdata(1000:1512,768:1362),sscale=16
window,3
show3,trec(768:1280,768:1280),sscale=16

;run again or quit
print,'type 99 to quit, any other number to continue'
iquit=1
read,iquit
if iquit eq 99 then goto, quitme
goto,loop
quitme:

close,1

;if iquit eq 99 then stop
end
```

C program to apply transfer function to grayscale images

```
#include <stdio.h>
#include <stdlib.h>
#include "main.h"


#define DPAUSE { char c; fflush(stdin); scanf("%c", &c); }

   resolution of data
#define RES 256

// image structs
AUX_RGBImageRec *raw_image;
AUX_RGBImageRec *t_image;

// adjustment val
```

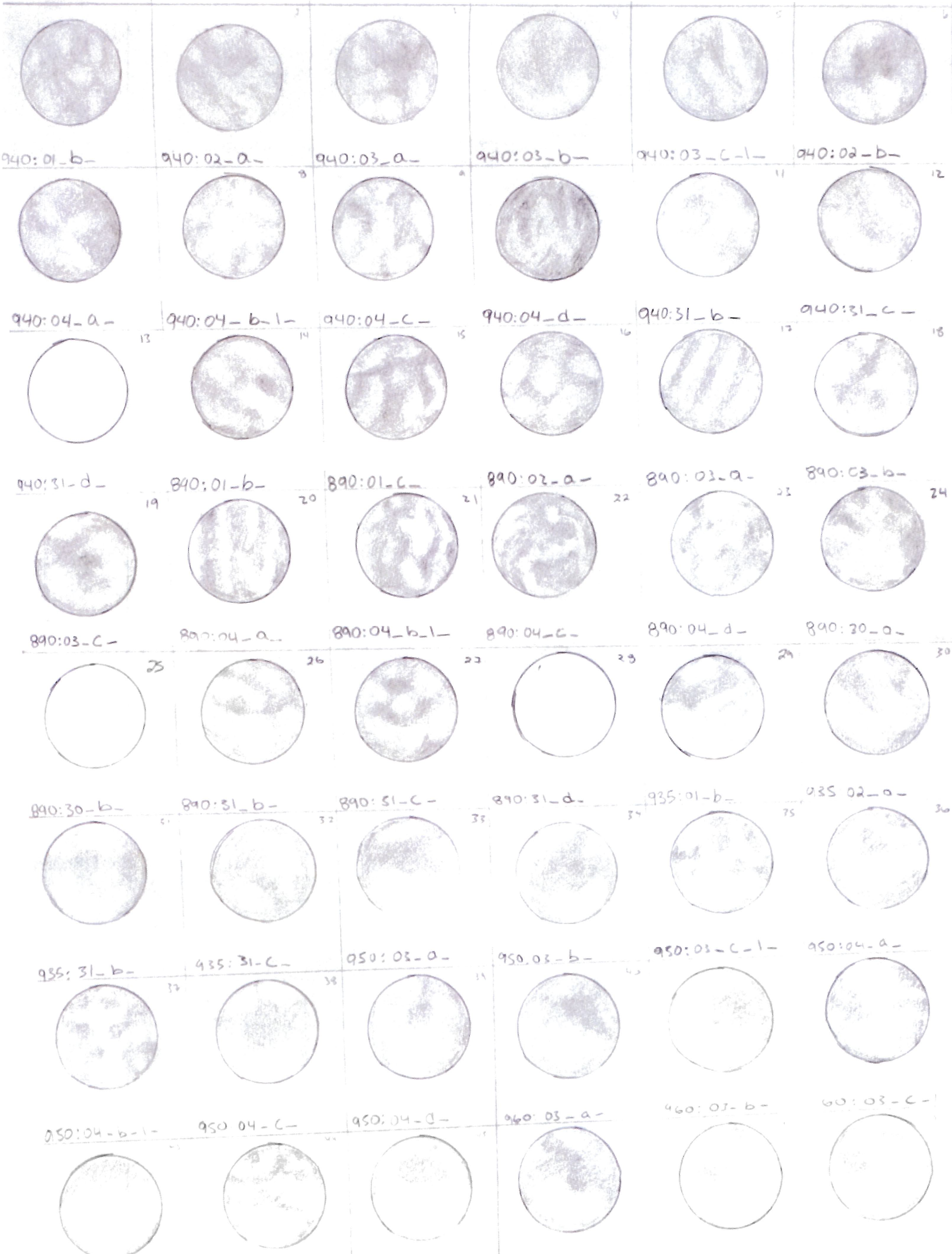
```
int aval;

// transfer function - maps [0.255] to RGB (0.255)
int tfunc_red[RES];
int tfunc_green[RES];
int tfunc_blue[RES];

int main(int argc, char **argv)
{
	int i;

	if (argc != 2) {
		printf("Usage: ... filename.bmp\n");
		exit(1);
	}

	// load the bitmap
	raw_image = LoadBMP(argv[1]);

	// make a copy of the image
	t_image = (AUX_RGBImageRec *) malloc(sizeof(AUX_RGBImageRec));
	t_image->sizeX = raw_image->sizeX;
	t_image->sizeY = raw_image->sizeY;
	t_image->data = (unsigned char *) malloc(t_image->sizeX * t_image->sizeY * 3 * sizeof(unsigned char));
	for (i=0; i<(t_image->sizeX * t_image->sizeY * 3); i++)  t_image->data[i] = raw_image->data[i];

	// init the adjustment val
	aval = 128;

	// init the transfer function
	tfunc_init();

	// apply the transfer function
	tfunc_apply();

	// register callbacks
	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_RGB | GLUT_DOUBLE);
	glutInitWindowSize(raw_image->sizeX,raw_image->sizeY);
	glutInitWindowPosition(100,100);
	glutCreateWindow(argv[0]);
	glutDisplayFunc(display);
	glutReshapeFunc(reshape);
	glutSpecialFunc(special);
	glutMainLoop();

	return 0;
}

void special(int key, int x, int y)
{

	// interactive adjustment of transfer function
	switch(key) {
		case GLUT_KEY_UP:
			aval+=1;
			printf("aval: %i     \r", aval);
			tfunc_init();
			tfunc_apply();
			break;
		case GLUT_KEY_DOWN:
			aval-=1;
```

```
                        printf("aval: %i    \r", aval);
                        tfunc_init();
                        tfunc_apply();
                        break;
                default:
                        break;
        }

        glutPostRedisplay();
}

void tfunc_apply()
{
        int i, indexval;

        // apply the transfer function to the copied image
        for (i=0; i<(raw_image->sizeX * raw_image->sizeY * 3); i+=3) {
                indexval = raw_image->data[i];
                t_image->data[i]   = tfunc_red[indexval];
                t_image->data[i+1] = tfunc_green[indexval];
                t_image->data[i+2] = tfunc_blue[indexval];
        }

}

void tfunc_init()
{
        int i, j, nval;

        // init all to zero
        for (j=0; j<RES; j++) {
                tfunc_red[j] = 0;
                tfunc_green[j] = 0;
                tfunc_blue[j] = 0;
        }

        // set the red channel
        for (i=0; i<256; i++) {
                nval = aval - abs(128-i);
                if (nval < 0) nval = 0;
                if (nval > 255) nval = 255;
                tfunc_red[i] = nval;
        }

        // set the green channel
        for (i=0; i<256; i++) {
                nval = aval - abs(128-i);
                if (nval < 0) nval = 0;
                if (nval > 255) nval = 255;
                tfunc_green[i] = nval;
        }

        // set the blue channel
        for (i=128; i<255; i++) {
                tfunc_blue[i] = i-128;
        }

}

void display(void)
{
        glClear(GL_COLOR_BUFFER_BIT);
```

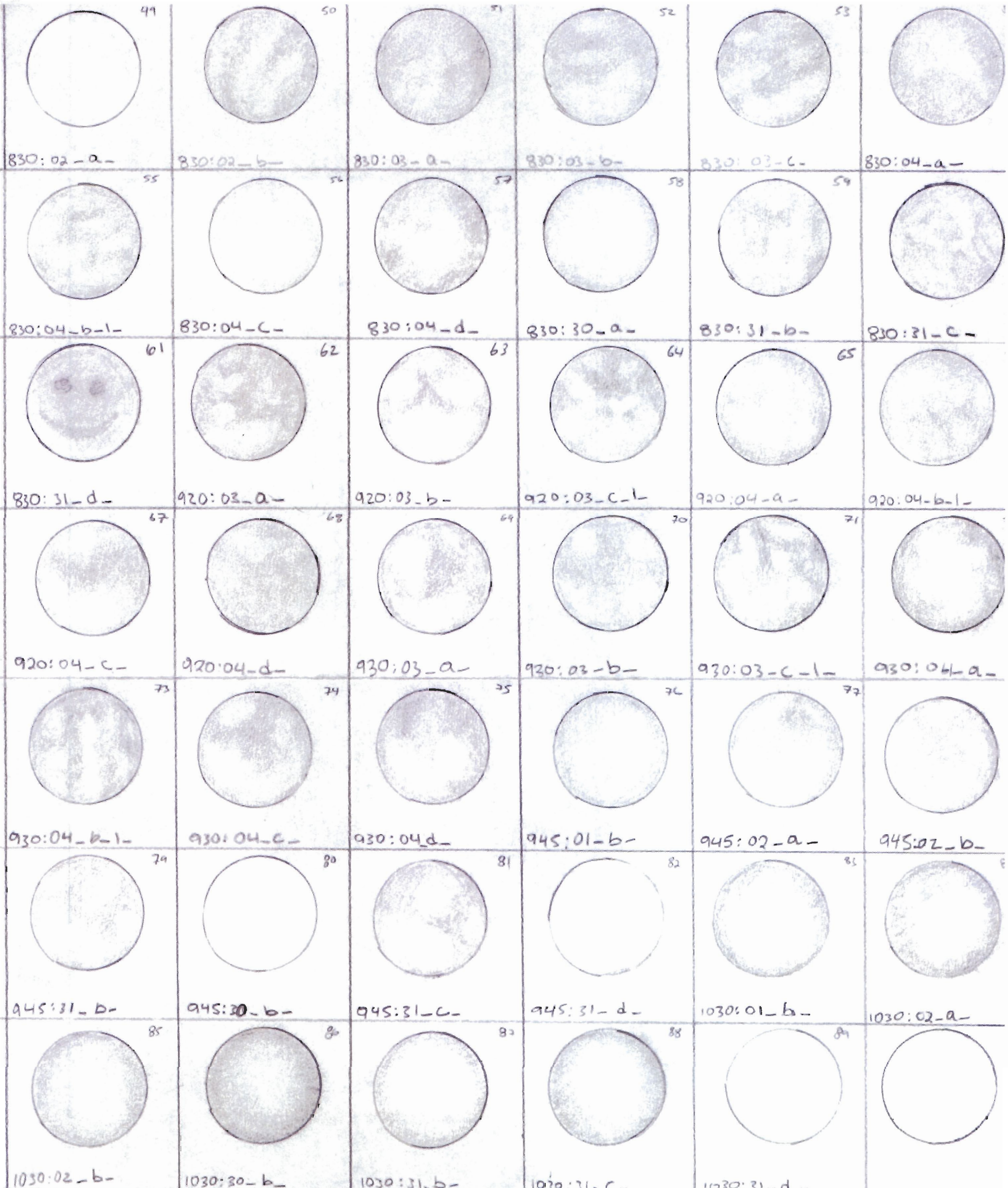
```
        glRasterPos2i(0,0);

        glDrawPixels(raw_image->sizeX, raw_image->sizeY, GL_RGB, GL_UNSIGNED_BYTE, t_image->data);
        glutSwapBuffers();
}

void reshape(int w, int h)
{

        glMatrixMode(GL_PROJECTION);
        glLoadIdentity();
        gluOrtho2D(0.0, (float) w, 0.0, (float) h);
        glMatrixMode(GL_MODELVIEW);
        glLoadIdentity();
}

  BMP loader function
AUX_RGBImageRec *LoadBMP(char *Filename)                    // Loads A Bitmap Image
{
        FILE *File=NULL;                                    // File Handle

        if (!Filename)                                      // Make Sure A Filename Was Given
        {
                return NULL;                                // If Not Return NULL
        }

        File=fopen(Filename,"r");                           // Check To See If The File Exists

        if (File)                                           // Does The File Exist
        {
                fclose(File);                               // Close The Handle
                return auxDIBImageLoad(Filename);           // Load The Bitmap And Return A Pointer
        }

        return NULL;                                        // If Load Failed Return NULL
}
```

A.2

This section includes sketches of all the images studied for the project, from which the best of each wavelength were chosen to be included in this paper.



940:01_b_
940:02_a_
940:03_a_
940:03_b_
940:03_c_1_
940:02_b_
940:04_a_
940:04_b_1_
940:04_c_
940:04_d_
940:31_b_
940:31_c_
940:31_d_
840:01_b_
890:01_c_
890:02_a_
890:03_a_
890:03_b_
890:03_c_
890:04_a_
890:04_b_1_
890:04_c_
890:04_d_
890:30_a_
890:30_b_
890:31_b_
890:31_c_
890:31_d_
935:01_b_
935 02_a_
935:31_b_
935:31_c_
950:03_a_
950:03_b_
950:03_c_1_
950:04_a_
950:04_b_1_
950 04_c_
950:04_d_
960:03_a_
960:03_b_
60:03_c_


49
830:02_a_
50
830:02_b_
51
830:03_a_
52
830:03_b_
53
830:03_c_
830:04_a_
55
830:04_b_1_
56
830:04_c_
57
830:04_d_
58
830:30_a_
59
830:31_b_
830:31_c_
61
830:31_d_
62
920:03_a_
63
920:03_b_
64
920:03_c_1_
65
920:04_a_
920:04_b_1_
67
920:04_c_
68
920:04_d_
69
930:03_a_
70
930:03_b_
71
930:03_c_1_
930:04_a_
73
930:04_b_1_
74
930:04_c_
75
930:04_d_
76
945:01_b_
77
945:02_a_
945:02_b_
79
945:31_b_
80
945:30_b_
81
945:31_c_
82
945:31_d_
83
1030:01_b_
1030:02_a_
85
1030:02_b_
86
1030:30_b_
87
1030:31_b_
88
1030:31_c_
89
1030:31_d_

A.3

This section contains the parameters used in the IDL program to enhance the images.

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
920	3	a	a	2.3/.01	2.3/0	62
		b	b	3.4/.01	3.3/0	63
		c	x	x	x	x
		c_1_	c_1_	4.0/.01	4.0/0	64
		d	x	x	x	x
	4	a	a	3.0/.01	2.8/0	65
		b_1_	b_1_	4.7/.01	4.5/0	66
		c	c_1_	2.9/.01	2.9/0	67
		d	d	3.1/.01	2.9/0	68

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
930	3	a	a	2.5/.01	2.5/0	69
		b	b	3.5/.01	3.3/0	70
		c	x	x	x	x
		c_1_	c_1_	4.3/.01	4.3/0	71
		d	x	x	x	x
	4	a	a	3.3/.01	3.3/0	72
		b_1_	b_1_	4.1/.01	4.1/0	73
		c	c_1_	2.6/.01	2.6/0	74
		d	d	3.4/.01	3.3/0	75

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
935	1	a	x	2.7/.01	2.4/0	29
		b	a	x	x	x
		c	b	3.2/.01	2.8/0	30
	2	a	a	x	x	x
		b	x	x	x	x
	30	a	x	x	x	x
		x	a_1_	x	x	x
		b	x	x	x	x
		x	b_1_	x	x	x
	31	a	x	x	x	x
		b	a	3.6/.01	3.3/.01	31
		c	a	3.5/.01	3.2/0	32
		d	b	3.3/.01	?	33

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
940	1	a	x	x	x	x
		b	a	3.7/.01	3.0/.01	1
		c	x	x	x	x
	2	a	a	3.1/.01	2.7/.01	2
		b	b	3.0/.01	2.9/0	6
	3	a	a	2.7/.01	2.5/0	3
		b	b	3.7/.01	3.4/0	4
		c	x	x	x	x
		c_1_	c_1_	4.4/0	4.4/0	5
		x	_c1_	x	x	x
		d	x	x	x	x
	4	a	a	3.6/.01	3.3/0	7
		b_1_	b_1_	4.7/0	4.5/0	8
		x	_b1_	x	x	x
		c	c_1_	2.7/.01	2.5/0	9
		d	d	2.6/.01	2.3/0	10
	30	a	x	x	x	x
		x	_a1_	x	x	x
		x	a_1_	x	x	x
		b	x	x	x	x
		x	_b1_	x	x	x
		x	b_1_	x	x	x
	31	a	x	x	x	x
		b	a	3.7/.01	3.4/0	11
		c	a	3.2/.01	3.2/0	12
		d	b	3.3/.01	?	13

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
950	3	a	a	3.0/.01	2.9/0	33
		b	b	3.0/.01	3.0/0	34
		c	x	x	x	x
		c_1_	c_1_	3.9/.01	3.9/0	35
		d	x	x	x	x
	4	a	a	3.0/.01	2.7/0	36
		b_1_	b_1_	4.0/.01	4.0/0	37
		c	c	2.5/.01	2.3/0	38
		d	d	3.5/.01	3.0/0	39

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
960	3	a	a	3.3/.01	3.1/0	40
		b	b	3.7/.01	3.7/0	41
		c	x	x	x	x
		c_1_	c_1_	4.5/.01	4.5/0	42
		d	x	x	x	x
	4	a	a	3.5/.01	3.5/0	43
		b_1_	b_1_	4.2/.01	4.2/0	44
		c	c	3.5/.01	3.5/0	45
		d	d	2.9/.01	2.9/0	46

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
945	1	a	x	x	x	x
		b	a	3.7/.01	3.7/0	76
		c	x	x	x	x
	2	a	a	2.9/.01	2.7/0	77
		b	b	3.4/.01	3.2/0	78
	30	a	x	x	x	x
		x	a_1_	x	x	x
		b	b	?	?	80
	31	a	x	x	x	x
		b	a	3.0/.01	2.9/0	79
		c	a	3.3/.01	3.1/0	81
		d	b	?	?	82

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
1030	1	a	x	x	x	x
		b	a	5.2/.01	5.2/0	83
		c	x	x	x	x
	2	a	a	5.0/.01	5.0/0	84
		b	b	6.5/.01	6.5/0	85
	30	a	x	x	x	x
		x	a_1_	x	x	x
		b	b	10/.01	10.0/0	86
	31	a	x	x	x	x
		b	a	5.6/.01	5.6/0	87
		c	a	5.9/.01	5.9/0	88
		d	b	?	?	89

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
890	1	a	x	x	x	x
		b	a	3.4/.01	3.1/0	14
		c	b	2.3/.01	2.1/0	15
	2	a	a	3.0/.01	2.7/0	16
		b	x	x	x	x
	3	a	a	3.0/.01	3.0/0	17
		b	b	3.0/.01	2.7/0	18
		c	c	3.2/.01	2.9/0	19
		c_1_	x	x	x	x
		d	x	x	x	x
	4	a	a	2.7/.01	2.6/0	20
		x	b	x	x	x
		b_1_	b_1_	3.9/.01	3.8/0	21
		c	c	2.4/.01	2.2/0	22
		d	d	3.2/0.01	2.8	23
	30	a	a	3.0/.01	3.2/0	24
		b	b	9.0	?	25
	31	a	x	x	x	x
		b	a	2.8/.01	2.5/0	26
		c	a	3.0/.01	2.7/0	27
		d	b	3.0/.01	?	28

wavelength	date	H (fits)	T(fits)	H(fwhm/clipping)	T(fwhm/clipping)	#
830	1	a	x	x	x	x
		b	a	3.1/.01	2.9/0	47
		c	b	3.1/.01	2.9/0	48
	2	a	a	2.0/.01	noisy	49
		b	b	3.0/.01	2.9/0	50
	3	a	a	2.9/.01	2.9/0	51
		b	b	3.2/.01	3.1/0	52
		c	c	2.8/.01	2.8/0	53
		c_1_	x	x	x	x
		d	x	x	x	x
	4	a	a	2.6/.01	2.6/0	54
		b_1_	b_1_	4.2/.01	4.2/0	55
		c	c	3.1/.01	2.8/0	56
		d	d	2.3/.01	2.2/0	57
	30	a	a	3.7/.01	3.7/0	58
		b	x	x	x	x
	31	a	x	x	x	x
		b	a	2.4/.01	2.3/0	59
		c	a	2.4/.01	2.3/0	60
		d	b	2.4/.01	off	61