Additional File 6: High rate acquisition code

Script H1: Check acquisition, export images

```
function checkmikrospec8
%This Matlab function manages an incoming image stream from Mikrospec
% RT_Multi thermal image acquisition program.  Incoming image are put in
% home directory, and this function checks periodically to make sure
% things are running, and also makes jpg images of the thermal images to
% send to an internal webserver
%
% Matthew R. Patrick
% Hawaiian Volcano Observatory
% US Geological Survey
%
% June 26, 2013
%
% to automate this, put the following line in a batch file and run the batch
% file at specified intervals using Windows Scheduler:
%
% matlab -automation -r checkmikrospec
%
% Note: for the restarting of Mikrospec to work correctly, you need to have PTFB
% Pro running (and trained to hit the "Run" and "OK" buttons on the windows
% that subsequently appear upon the program start

cd('C:\Mikron'); %cd to directory where Mikrospec images are incoming
wm=imread('usgslogo_white.jpg'); %usgs logo for image overlay

y1=datestr(now,'yyyy'); %get current year

cd(['H:\MikronAcquire\' y1 '\Camera01']); %cd to current year

d=dir('*rti'); %list all Mikrospec images in this folder

lastimage=0;
for i=1:length(d) %find time of most recent image
    if d(i).datenum>lastimage
        lastimage=d(i).datenum;
    end
end

timediff=(now-lastimage)*86400; %time since last image in seconds

if timediff>120 %if last image is older than two minutes...
    dos('taskkill /F /IM MikroSpec_RT.exe'); %kill Mikrospec
    %restart Mikrospec and run in background:
    dos('"C:\Program Files (x86)\RT_Multi\MikroSpec_RT.exe" 0 &');
    dos('exit');
end

%make web image
h=figure(1);
set(h,'Units','inches')
set(h,'Position',[0 0 8 6])
set(h,'PaperPosition',[0 0 8 6])

if length(d)==0
    exit
end
```

```
FID=fopen(d(end).name,'r+','ieee-le');%imports file
A=fread(FID,[320,240],'short','ieee-le');%reads file
fclose(FID);
A=A/10; %get to kelvin
A=A-273.15; %convert to celsius
A=rot90(A,-1); %rotate image
A=fliplr(A); %flip image
A=[A(:,11:320) A(:,1:10)]; %rearrange image correctly
A=double(A);
m=d(end).name(1:2); %month
dd=d(end).name(4:5); %day
yy=d(end).name(7:10); %year
hh=d(end).name(12:13); %hour
mm=d(end).name(15:16); %minute
ss=d(end).name(18:19); %second

Acut=A(20:end,:);
minA=min(Acut(:));
maxA=max(Acut(:));
ah=axes('Position',[0 0 1 1]);
imagesc(A,[20 0.95*maxA]); %display image
hold on
colorbar
axis off
axis equal
axis tight
set(h,'Position',[0 0 8.8 6])
set(h,'PaperPosition',[0 0 8.8 6])
load iron %iron is an optional colormap, jet also works
colormap(iron)
%timestamp:
text(10,10,['HTcam: ' m '-' dd '-' yy ' ' hh ':' mm ':' ss ' HST'],...
    'fontsize',10,'BackgroundColor','white','Margin',2);
axis tight
image(wm,'Xdata',[2 27],'Ydata',[234 239]); %put on USGS logo overlay

line1=['var datetime = "' yy '-' m '-' dd ' ' hh ':' mm ':' ss ' (HST)";'];
line2='var frames   = new Array("M");';

try
    cd('C:/Mikron/webserver');
    print('-r150','-djpeg85','M.jpg'); %write image to file
    print('-r45','-djpeg85','M.thumb.jpg')%write image to file
    fid=fopen('js.js','wt');
    fprintf(fid, '%s\n', line1);
    fprintf(fid, '%s\n', line2);
    st=fclose(fid);
catch exception
    exception
end
try
    %copy image to webserver computer
    dos('robocopy C:\Mikron\webserver L:\cams\HTcam\images /COPY:DT /R:5');
catch exception
end

%=====================================================
%put images in kala webserver for internal webpage
%=====================================================
try
    year=yy;
    month=m;
    day=dd;
```

```
hour=hh;
mint=mm;
sec=ss;
thistime=now;
% make one jpg image per minute for last five minutes
for i=0:4
    cd(['H:\MikronAcquire\' y1 '\Camera01']);
    d=dir('*rti');

    t=thistime-(i*1/1440);
    mindiff=300;
    for k=1:length(d);
        m=d(k).name(1:2);
        dd=d(k).name(4:5);
        yy=d(k).name(7:10);
        hh=d(k).name(12:13);
        mm=d(k).name(15:16);
        ss=d(k).name(18:19);
        thist=datenum(str2num(yy),str2num(m),str2num(dd),str2num(hh)...
            ,str2num(mm),str2num(ss));
        timediff=abs(t-thist);
        if timediff<mindiff
            mindiff=timediff;
            bestindex=k;
        end

    end

    if mindiff<(1/1440)
        %open image and display
        m=d(bestindex).name(1:2);
        dd=d(bestindex).name(4:5);
        yy=d(bestindex).name(7:10);
        hh=d(bestindex).name(12:13);
        mm=d(bestindex).name(15:16);
        ss=d(bestindex).name(18:19);

        year=yy;
        month=m;
        day=dd;
        hour=hh;
        mint=mm;
        sec=ss;

        FID=fopen(d(bestindex).name,'r+','ieee-le');%imports file
        A=fread(FID,[320,240],'short','ieee-le');%reads file
        fclose(FID);

        A=A/10; %get to kelvin
        A=A-273.15; %convert to celsius
        A=rot90(A,-1); %rotate image
        A=fliplr(A); %flip image
        A=[A(:,11:320) A(:,1:10)];
        A=double(A);

        Acut=A(20:end,:);
        minA=min(Acut(:));
        maxA=max(Acut(:));

        h=figure(1); %set up figure window
        set(h,'Units','inches')
        set(h,'Position',[0 0 8 6])
        set(h,'PaperPosition',[0 0 8 6])
```

```
clf

ah=axes('Position',[0 0 1 1]);
imagesc(A,[20 0.95*maxA]); %display array as image
colorbar
axis off
axis equal
axis tight
set(h,'Position',[0 0 8.8 6])
set(h,'PaperPosition',[0 0 8.8 6])
load iron %iron is optional colormap, jet also works
colormap(iron)
%timestamp:
text(10,10,['HTcam: ' m '-' dd '-' yy ' ' hh ':' mm ':' ss ...
    ' HST'],'fontsize',10,'BackgroundColor','white','Margin',2);
axis tight

cc='G:/HTcam/';
cd(cc)
d=dir;
%check if year folder exists
key=0;
for j=1:length(d)
    dyear=d(j).name;
    if strcmp(year,dyear)==1 & d(j).isdir==1
        key=1;
    end
end
if key==1
    cd(year);
else mkdir(year)
    cd(year);
end

%check if month folder exists
key=0;
d=dir;
for j=1:length(d)
    dmonth=d(j).name;
    if strcmp(month,dmonth)==1 & d(j).isdir==1
        key=1;
    end
end
if key==1
    cd(month);
else mkdir(month)
    cd(month);
end

%check if day folder exists
d=dir;
key=0;
for j=1:length(d)
    dday=d(j).name;
    if strcmp(day,dday)==1 & d(j).isdir==1
        key=1;
    end
end
if key==1
    cd(day);
else mkdir(day)
    cd(day)
```

```
            end

            %uncheck section below if using hour folders:

            %check if hour folder exists
            d=dir;
            key=0;
            for j=1:length(d)
                j;
                dhour=d(j).name;
                if strcmp(hour,dhour)==1 & d(j).isdir==1
                    key=1;
                end
            end
            if key==1
                cd(hour);
            else mkdir(hour)
            end

            %go back to root directory
            cd(cc);
            cd(year);
            cd(month);
            cd(day);
            cd(hour);
            %uncomment line below if using hour folders,
             %and comment out line after that
            print('-r150','-djpeg85',[year month day hour mint sec 'M.jpg']);

            dx=dir([year month day hour mint sec 'M.jpg']);
            try
                if dx(1).bytes>10000 %if image is not corrupt
                    %copy images to webserver computer
                    dos(['robocopy G:\HTcam\' year '\' month '\' day '\' ...
                        hour '\ L:\cams\HTcam\images\archive\' year '\' ...
                        month '\' day '\' hour '\ /COPY:DT /R:1 /E /XF Thumbs.db']);
                end
            catch exception
                exception
            end
        end
    end
catch exception
    exception
end

exit %exit Matlab
```

Script H2: Run scripts H2a-d

```
function runprograms3
% Matlab function is the master program run each hour to analyze and organize
% incoming thermal images from the high rate acquisition.
%
% The main steps:
% 1) tempfilemove: move images from the Mikrospec output folder to a
% temporary folder for analysis
% 2) lavalevelMikron_7run4: measure lava level in these images
% 3) crustvelocityMikron_test4: measure crust velocity
% 4) temperaturealarm: run temperature alarm if needed
% 5) moveIRfiles_mycomputer2: once images are analyzed, move them from temp
% folder to date based folder structure for archiving
%
% Matthew R Patrick
% US Geological Survey
% Hawaiian Volcano Observatory
%
% June 26, 2013
%

% move images from Mikrospec output folder to temporary folder for analysis
tempfilemove

cd('H:\Mikron\');

d=dir;
if length(d)<2000 %if less than 2000 images (i.e. if manageable number)
    try
        leveltest05e2
    catch exception
    end
end

%crustvelocityMikron_test4
%temperaturealarm

%move files to date based folders for archiving
moveIRfiles_mycomputer2

exit %exit Matlab
```

Script H2a: Move incoming images to temporary directory for analysis

```
function tempfilemove
% Matlab function moves files from the output directory used by Mikrospec_RT
% to a temporary folder where files are available for analysis.  The
% function is meant to run once an hour, and this function is run by
% runprograms3.m.
%
% Matthew R Patrick
% US Geological Survey
% Hawaiian Volcano Observatory
%
% June 26, 2013
%

y1=datestr(now,'yyyy'); %get current year

cd(['H:\MikronAcquire\' y1 '\Camera01']); %go to Mikrospec output directory
d1=dir('*jpg');
if length(d1)>0
    movefile('*jpg','H:\Mikron'); %move to temp directory
end
d2=dir('*rti');
if length(d2)>0
    movefile('*rti','H:\Mikron'); %move to temp directory
end
```

Script H2b: Measure lava lake level

```
function lavalevelMikron_7run3
%This Matlab function analyzes incoming thermal images of the lava lake in
%Halemaumau Crater at Kilauea Volcano and locates the boundary of the lava
%lake against the back wall, providing a relative measurement of lava
%level.  This script is run by Script H2.
%
% Matthew R Patrick
% Hawaiian Volcano Observatory
% US Geological Survey
%
% July 4, 2013
%

%go to temporary folder where Script H2a places the images
cd H:\Mikron

%initialize
level=[];
alldates=[];
count=0;
Amax=zeros(240,320);
Aedgemax=zeros(240,320);
Aedgesum=zeros(240,320);

d=dir('*.rti');
for i=1:length(d)  %go through all images in folder
    count=count+1;
    FID=fopen(d(i).name,'r+','ieee-le');%imports file
    A=fread(FID,[320,240],'short','ieee-le');%reads file
    fclose(FID);
    A=A/10; %get to kelvin
    A=A-273.15; %convert to celsius
    A=rot90(A,-1); %rotate image
    A=fliplr(A); %flip image
    A=[A(:,11:320) A(:,1:10)]; %rearrange image
    A=double(A);

    Aedge=edge(A);  %construct edge image
    Aedge=double(Aedge);
    Aedgesum=Aedgesum+Aedge;  %sum up edge images
    %go pixel by pixel and determine max value in this location
    for r=1:size(A,1)
        for c=1:size(A,2)
            if A(r,c)>Amax(r,c)
                Amax(r,c)=A(r,c);
            end
        end
    end
    Acube(:,:,count)=A;  %construct 3d block of images
    %if this is the end of the four minute time block...
    if count==48
        count=0;
        %go through 3d block and determine pixel by pixel std
        %values
        for r=1:size(A,1)
            for c=1:size(A,2)
                Astd(r,c)=std(Acube(r,c,:));
            end
        end
```

```
        Amax=double(Amax);
        Amax2=(Amax-mean(Amax(:)))./std(Amax(:)); %normalize
        Astd2=(Astd-mean(Astd(:)))./std(Astd(:)); %normalize
        newA=5*Amax2+7*Astd2; %combine max and std images
        newA=(newA-mean(newA(:)))./std(newA(:)); %normalize

        se = strel('disk',1);
        Aedgesum=imclose(Aedgesum,se); %closing - filter
        Aedgesum=imopen(Aedgesum,se); %opening - filter
        Aedgesum=(Aedgesum-mean(Aedgesum(:)))./std(Aedgesum(:)); %normalize
        Aedgesum=Aedgesum./2;

        Athresh=(Amax>350)*10;  %make threshold image

        newA=newA+Aedgesum+Athresh;  %combine all four criteria

        Aline2=sum(newA(:,160:190)'); %take vertical swath along back wall
        Amax=zeros(240,320);
        Aedgemax=zeros(240,320);
        Aedgesum=zeros(240,320);
        linethresh=35;  %this is the threshold value for picking lava level
        key=0;
        %go through vertical swath and find where it exceeds
        %threshold
        for j=30:length(Aline2)
            if Aline2(j)>linethresh & key==0
                topj=j;
                key=1;
            end
        end

        %get date and time info from filename
        m=d(i).name(1:2);
        dd=d(i).name(4:5)
        yy=d(i).name(7:10);
        hh=d(i).name(12:13)
        mm=d(i).name(15:16)
        ss=d(i).name(18:19);
        %make matlab datenum for image time
        thist=datenum(str2num(yy),str2num(m),str2num(dd),str2num(hh),...
            str2num(mm),str2num(ss));
        %quality control factor: if max temp<300 ignore (too fumey)
        if max(max(A(10:220,10:310)))>300
            level=[level topj];
            alldates=[alldates thist];
        end
    end
end
level=medfilt1(level,5);  % 5 point median filter

%write text file of results formatted for Script H4
dates=alldates;
besterh=level+1;
cd('C:\Mikron\MikronTextFiles\leveltest\');
s=datestr(now,31);
x=[s(1:10) '_' s(12:13) s(15:16) '_level.txt'];
M=[alldates' besterh'];
if length(M)>0
    dlmwrite(x,M,'newline','pc','precision',13);
end
```

Script H2c: Measure lava lake surface velocity

```
function crustvelocityMikron_test4
% This Matlab function goes to a folder with an hour's worth of thermal
% images of the lava lake in Halemaumau and calculates the speed at which
% the surface crust is moving, using a very simple 2D cross correlation of
% a window taken at the center of the lava lake. This script writes the
% result to a text file.  This script is run by Script H2.
%
% Matthew R Patrick
% Hawaiian Volcano Observatory
% US Geological Survey
%
% Feb 11, 2011
%

%go to directory with binary thermal images
cd('H:\Mikron');

d=dir('*rti');

%initialize arrays
datesv=[];
level=[];
maxtemp=[];
meantemp=[];
jindex=[];
velocity=[];

profile=zeros(240,1);

%pixel size of .75 m divided by cos(40) %for 50 dg lens, approx. - this
%estimate is only rough and only applies to certain distance from lake to
%camera- future code needs to adjust this automatically based on lava level
pixelsize=0.979;
step=5;

%go through all images
for j=step+1:2:length(d)

    %open first image
    FID1=fopen(d(j-step).name,'r+','ieee-le');%imports file
    A=fread(FID1,[320,240],'short','ieee-le');%reads file
    A=A/10; %get to kelvin
    A=A-273.15; %convert to celsius
    A=rot90(A,-1); %rotate image
    A=fliplr(A); %flip image
    A=[A(:,11:320) A(:,1:10)];
    month=str2num(d(j-step).name(1:2));
    day=str2num(d(j-step).name(4:5));
    year=str2num(d(j-step).name(7:10));
    hour=str2num(d(j-step).name(12:13));
    minute=str2num(d(j-step).name(15:16));
    sec=str2num(d(j-step).name(18:19));
    date1=datenum(year,month,day,hour,minute,sec);

    %open second image
    FID2=fopen(d(j).name,'r+','ieee-le');%imports file
    B=fread(FID2,[320,240],'short','ieee-le');%reads file
    B=B/10; %get to kelvin
    B=B-273.15; %convert to celsius
```

```
    B=rot90(B,-1); %rotate image
    B=fliplr(B); %flip image
    B=[B(:,11:320) B(:,1:10)];
    month=str2num(d(j).name(1:2));
    day=str2num(d(j).name(4:5));
    year=str2num(d(j).name(7:10));
    hour=str2num(d(j).name(12:13));
    minute=str2num(d(j).name(15:16));
    sec=str2num(d(j).name(18:19));
    date2=datenum(year,month,day,hour,minute,sec);

    %time difference between images
    dT=(date2-date1)*86400;

    %make correlation window
    ydim=20;
    xdim=20;
    Ayc=130;
    Axc=155;
    Awin=A(Ayc-ydim/2:Ayc+ydim/2,Axc-xdim/2:Axc+xdim/2);

    %make search area
    Barea=B(110:180,110:210);

    %do 2D cross correlation
    C=normxcorr2(Awin,Barea);

    [max_cc, imax]=max(abs(C(:)));
    [ypeak, xpeak]=ind2sub(size(C),imax(1));
    ypeaksave(j)=ypeak;
    coeff(j)=max_cc; %save maximum correlation

    corr_center=[xpeak-size(Awin,1)/2+110 ypeak-size(Awin,2)/2+110];
    corr_offset=[corr_center(1)-155 corr_center(2)-130];
    totaldist=sqrt(sum(corr_offset.^2))*pixelsize;

    if coeff(j)>0.9
        datesv=[datesv date2];
        velocity=[velocity totaldist/dT];
    end

    st1=fclose(FID1);
    st2=fclose(FID2);
end

%write text file of results, first column time, second column velocity
cd('D:\Mikron\MikronTextFiles\CrustVelocity');
s=datestr(now,31)
format long g
x=[s(1:10) '_' s(12:13) s(15:16) '_velocity.txt']
M=[datesv' velocity'];

if length(M)>0
    dlmwrite(x,M,'newline','pc','precision',13);
end
```

Script H2d: Archive images in date folders

```
function moveIRfiles_mycomputer2
% Matlab function moves thermal image files from temporary folder
% to date based folder structure for long term archiving
%
% Matthew R Patrick
% US Geological Survey
% Hawaiian Volcano Observatory
%
% June 26, 2013
%

cd('H:\Mikron'); %temporary folder used for hourly analysis

d1=dir('*rti');
d2=dir('*jpg');

%go through rti files first
for i=1:length(d1)
    d=dir;
    month=d1(i).name(1:2);
    day=d1(i).name(4:5);
    year=d1(i).name(7:10);
    hour=d1(i).name(12:13);
    minute=d1(i).name(15:16);
    sec=d1(i).name(18:19);

    %check if year folder exists
    key=0;
    for j=1:length(d)
        j;
        dyear=d(j).name;
        if strcmp(year,dyear)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(year);
    else mkdir(year)
        cd(year);
    end

    %check if month folder exists
    key=0;
    d=dir;
    for j=1:length(d)
        j;
        dmonth=d(j).name;
        if strcmp(month,dmonth)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(month);
    else mkdir(month)
        cd(month);
    end

    %check if day folder exists
    d=dir;
```

```
    key=0;
    for j=1:length(d)
        j;
        dday=d(j).name;
        if strcmp(day,dday)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(day);
    else mkdir(day)
        cd(day)
    end

    %check if hour folder exists
    d=dir;
    key=0;
    for j=1:length(d)
        j;
        dhour=d(j).name;
        if strcmp(hour,dhour)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(hour);
    else mkdir(hour)
    end

    %go back to root directory
    cd('H:\Mikron');
    try
        %move file to correct folder
        movefile(d1(i).name,[year '/' month '/' day '/' hour '/' d1(i).name]);
    catch exception
    end
end
cd('H:\Mikron');
%now go through jpg files
for i=1:length(d2)
    d=dir;
    month=d2(i).name(1:2);
    day=d2(i).name(4:5);
    year=d2(i).name(7:10);
    hour=d2(i).name(12:13);
    minute=d2(i).name(15:16);
    sec=d2(i).name(18:19);

    %check if year folder exists
    key=0;
    for j=1:length(d)
        j;
        dyear=d(j).name;
        if strcmp(year,dyear)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(year);
    else mkdir(year)
        cd(year);
    end
```

```
    %check if month folder exists
    key=0;
    d=dir;
    for j=1:length(d)
        j;
        dmonth=d(j).name;
        if strcmp(month,dmonth)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(month);
    else mkdir(month)
        cd(month);
    end

    %check if day folder exists
    d=dir;
    key=0;
    for j=1:length(d)
        j;
        dday=d(j).name;
        if strcmp(day,dday)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(day);
    else mkdir(day)
        cd(day)
    end

    %check if hour folder exists
    d=dir;
    key=0;
    for j=1:length(d)
        j;
        dhour=d(j).name;
        if strcmp(hour,dhour)==1 & d(j).isdir==1
            key=1;
        end
    end
    if key==1
        cd(hour);
    else mkdir(hour)
    end

    %go back to root directory
    cd('H:\Mikron');
    pwd;
    try
        %move file to correct directory
        movefile(d2(i).name,[year '/' month '/' day '/' hour '/' d2(i).name]);
    catch exception
    end
end
```

Script H3: Make composite image

```
function mikroncomposite2
%Matlab function goes through thermal images over past day and creates
% pixel by pixel max image.  It converts this image to a jpg file and
% copies it to a webserver computer
%
% Note you need Robocopy installed for file transfer to work
%
% Matthew R Patrick
% Hawaiian Volcano Observatory
% US Geological Survey
%
% June 26, 2013
%

%go back 24 hours from now (or from chosen end)
%endtime=datenum('14 nov 2007 06:00:00');
endtime=now;
starttime=endtime-1;
startyear=datestr(starttime,10);
startmonth=datestr(starttime,5);
startday=datestr(starttime,7);
endyear=datestr(endtime,10);
endmonth=datestr(endtime,5);
endday=datestr(endtime,7);

Amax=zeros(240,320); %initial composite array

for d1=1:2 %go through previous day and current day
    if d1==1
        cd(['H:/Mikron/' startyear '/' startmonth '/' startday])
    elseif d1==2
        cd(['H:/Mikron/' endyear '/' endmonth '/' endday])
    end
    hrs=dir;
    if d1==1
        s=9;
    else s=3;
    end
    for i=s:length(hrs) %go through each day's nighttime hours
        if hrs(i).isdir==1
            cd(hrs(i).name);
            d=dir('*.rti'); %list all images in this frame
            for j=1:12:length(d) %go through all images in this frame
                d(j).name
                pwd
                %FID=fopen(d(j).name,'r+','ieee-le');%imports file
                FID=fopen(d(j).name,'r','ieee-le');%imports file
                A=fread(FID,[320,240],'short','ieee-le');%reads file
                fclose(FID);
                A=A/10; %get to kelvin
                A=A-273.15; %convert to celsius
                A=rot90(A,-1); %rotate image
                A=fliplr(A); %flip image
                A=[A(:,11:320) A(:,1:10)];
                A=double(A);

                for p=1:size(A,1) %go through each pix, find max
                    for q=1:size(A,2)
                        if A(p,q)>Amax(p,q)
```

```
                        Amax(p,q,1)=A(p,q,1);
                    end
                end
            end %end of max 2d loop
            m=str2num(d(j).name(1:2)); %month
            dd=str2num(d(j).name(4:5)); %day
            year=str2num(d(j).name(7:10)); %year
            hh=str2num(d(j).name(12:13)); %hour
            mm=str2num(d(j).name(15:16)); %minute
        end %end of each image
    end %end of if statement checking to see if hour folder exists
    cd ..
  end %end of each hour
end %end of each day

h=figure(1); %setup figure window
set(h,'Units','inches')
set(h,'Position',[0 0 8 6])
set(h,'PaperPosition',[0 0 8 6])

yy=datestr(now,10); %year
m=datestr(now,5); %month
dd=datestr(now,7); %day
hh=datestr(now,'HH'); %hour
mm=datestr(now,'MM'); %minute
ss=datestr(now,'SS'); %second

Acut=Amax(20:235,:);
minA=min(Acut(:)); %get min value for colorscale
maxA=max(Acut(:)); %get max value for colorscale
ah=axes('Position',[0 0 1 1]);
imagesc(Amax,[10 0.95*maxA]);
axis off
axis equal
axis tight
colorbar
set(h,'Position',[0 0 8.8 6])
set(h,'PaperPosition',[0 0 8.8 6])

load iron %iron is optional colormap, jet also good
colormap(iron)
%timestamp image:
text(10,10,[m '-' dd '-' yy ' ' hh ':' mm ':' ss ' HST Composite'],...
    'fontsize',10,'BackgroundColor','white','Margin',2);

line1=['var datetime = "' yy '-' m '-' dd ' ' hh ':' mm ':' ss ' (HST)";'];
line2='var frames   = new Array("M");';

try
    cd('C:/Mikron/composites/');
    print('-r150','-djpeg85','M.jpg'); %make image file
    fid=fopen('js.js','wt'); %write accessory file
    fprintf(fid, '%s\n', line1);
    fprintf(fid, '%s\n', line2);
    st=fclose(fid);
catch exception
    exception
end

try
    %copy image file to webserver computer
    dos('robocopy C:\Mikron\composites\ L:\cams\HTcam\composites /COPY:DT /R:1');
catch exception
```

```
end

year=yy;
month=m;
day=dd;

cc='C:/Mikron/composites';
cd(cc)
d=dir;
%check if year folder exists
key=0;
for j=1:length(d)
    dyear=d(j).name;
    if strcmp(year,dyear)==1 & d(j).isdir==1
        key=1;
    end
end
if key==1
    cd(year);
else mkdir(year)
    cd(year);
end

%check if month folder exists
key=0;
d=dir;
for j=1:length(d)
    dmonth=d(j).name;
    if strcmp(month,dmonth)==1 & d(j).isdir==1
        key=1;
    end
end
if key==1
    cd(month);
else mkdir(month)
    cd(month);
end

%go back to root directory
cd(cc);
cd(year);
cd(month);

%uncomment line below if using hour folders, and comment out line after that
print('-r150','-djpeg85',['HTcam' year month day 'M.jpg']); %make jpg image
%movefile(Aname2,[year '/' month '/' day '/' hour '/' Aname2]);
%movefile(Aname2,[year '/' month '/' day '/' Aname2]);

dx=dir(['HTcam' year month day 'M.jpg']);

try
    if dx(1).bytes>10000 %if image not corrupt
        dos(['robocopy C:\Mikron\composites\' year '\' month ...
            '\ L:\cams\HTcam\composites\archive\' year '\' month ...
            '\ /COPY:DT /R:1 /E /XF Thumbs.db']);
    end
catch exception
    exception
end

exit %exit Matlab
```

Script H4: Write data to VALVE webserver

```
function writevalvedata2
%Matlab function takes text files of lava level (two column data, first
%column is datenumber and second column is lava level in pixels), then does
%a mean filter and writes it to a new text file that is properly formatted
%for VALVE (HVOs internal dataserver).
%
% Matthew R Patrick
% US Geological Survey
% Hawaiian Volcano Observatory
%
% June 23, 2013
%

cd('C:\Mikron\MikronTextFiles\LavaLevel'); %go to folder with text files

d=dir('*.txt');

%get most recent text file
s=d(end).name;
t=d(end).datenum

threshold=2/24; %two hours
difft=now-t; %time since last text file

%if most recent text file is older than 2 hours, quit
if difft>threshold
    exit
end

a=dlmread(s,',');
acdate=a(:,1); %first column is date
level=a(:,2); %second column is lava level in pixels from top
level=240-level; %measure level from bottom of image

%apply mean filter
meanlevel=level;
for i=6:length(level)-5
    count=0;
    thissum=0;
    for j=i-5:i+5
        if level(j)>0
           thissum=thissum+level(j);
        count=count+1;
        end
    end
    meanlevel(i)=thissum./count;
end

%set up output array for text file
gmtdate=acdate+(10/24); %set to UTC time
ds=datestr(gmtdate,31);
if size(level,1)>size(meanlevel,2)
    val=num2str(meanlevel);
else val=num2str(meanlevel');
end

%write VALVE formatted text file
fid=fopen([datestr(now,'yyyymmddHHMM') 'HMM.txt'],'w');
for i=1:size(ds,1)
```

```
    fileline=[ds(i,:) ', HMM, ' val(i,:)];
    count=fprintf(fid, '%s\n', fileline);
end
st=fclose(fid);

s1=[datestr(now,'yyyymmddHHMM') 'HMM.txt'];
s2=['L:\valve3\geology\lavalevel\' datestr(now,'yyyymmddHHMM') 'HMM.txt'];

movefile(s1,s2); %move file

exit %exit Matlab
```

Script H5: Check harddrive space

```
function freespacealert
%Matlab function checks drive storing thermal images and sends out text
% message if free space on drive is getting low
%
% Matthew R Patrick
% Hawaiian Volcano Observatory
% US Geological Survey
%
% June 25, 2013
%
% The function that sends the text message (send_text_message.m) was
% written by Ke Feng (jnfengke@gmail.com) (Revision: 1.0.0.0, Date:
% 2007/09/28) and available on Matlab Central. You may be able to write
% your own simple text messaging code, considering that text messages can
% be sent like emails and you can use the Matlab "sendmail.m" function.
% If the mobile phone number is 012-345-6789 then the email address would
% be: 0123456789@vtext.com, where vtext.com is the domain used by Verizon,
% for example.  Other cell providers domains can be found easily on the web.

FileObj = java.io.File('H:\'); %java function
a=FileObj.getFreeSpace

a=a/1e9; %convert to Gb
a=round(a*10);
a=a/10;
s=['Only ' num2str(a) ' Gb left on H drive!'];
%if less than 8 Gb left, send text message
if a<8
    %insert mobile phone number in Xs and change phone company if needed
    send_text_message('XXX-XXX-XXXX','verizon','HTcam',s)
end
exit %exit Matlab
```