## Additional File 7: Low rate acquisition code

*Script L1: Acquire single image, export image*

```
function AcquireMikronImage_blade01
%function AcquireMikronImage_blade01
%
%This function acquires a single image (in three different formats) from a
% Mikron M7500 thermal camera using the ActiveX control offered by RT Multi
% software.  This program is designed to be run by a scheduler (e.g. Windows
% Scheduled Tasks) to acquire images at a frequency of your
% choosing.  Here is a step-by-step outline of what the program does:
%
% 1) Pings camera 10 times to wake it up
% 2) Connects to camera using ActiveX control in a Visual Basic .NET .exe
% program
% 4) Acquires binary (.bin) image - this is a generic format good for
% long-term archiving, but is harder for quick analyses
% 5) Makes .JPG image for archiving - good for a "quick look" of what's
% going on
% 6) Puts .bin, and .JPG files in appropriate date folder (year,
% month, day) - will make folders if necessary
% 7) Does simple temperature measurement (max in image, and temperature at
% three spots), writes this to a running text file
% 8) Makes .jpg file and puts into an html directory, and constantly
% updates this file.  This will allow users to have a web page with a
% constantly updated, current image from the camera - like a low-bandwidth
% version of FLIR's IR Monitor software
% 9) Delete any incorrectly acquired files
%10) Automatically exit Matlab
%
%====================================================================
% What YOU need to do:
%====================================================================
% 0) Of course, install the Mikron Multi RT software (which has the ActiveX control
% used in this program), and setup the camera on the network. The you have
% to go through the troublesome task of writing a Visual Basic .NET script
% that runs the ActiveX control (that requires much more description). The
% Visual Basic .NET program will be a .exe file that this Matlab script calls
% below.
%
% 1) Make a folder for the current camera, such as C:\Mikron\Cam01
%
% 2) In Part 1, put the image coordinates of three spots that you would
% like to have the temperature monitored at (this can easily be modified by
% someone comfortable with Matlab to add more points, or make the spots
% boxes)
%
% 3) In Part 2, below, insert the correct IP address, time zone (of the
% clock of the computer doing the acquisition) and folder name
%
% 4) If using Windows, make a .bat file that will run this Matlab function
% Make a Windows Scheduled Task to run the .bat file
% at a frequency of your choosing (e.g. every 10 minutes).
%
% 5) Setup the html file that will show the current image
%
%6) Sit back and be amazed at the images that are coming in!
%====================================================================
%
% Oh, and this is important:
```

```
% This function is intended for relatively slow acquisition rates - say
% every few minutes.  Because this function takes about 30 seconds
% to run, I would not choose your acquisition frequency to be any faster than
% 1 minute.  If you need to do high frame rate grabs, there are better ways
% to do it and I can guide you in those directions.
%
%
% Matthew R. Patrick
% Hawaiian Volcano Observatory
% US Geological Survey
%
%
% April 18, 2011


%=========================================================================
%Part 1: Designate measurement points: MODIFY THESE!!!!!!!!!!!!!!!!!!!!!!!!
%=========================================================================

%first measurement spot (row, column)
r1=160;
c1=120;
%second measurement spot
r2=100;
c2=100;
%third measurement spot
r3=200;
c3=200;

%=========================================================================
%Part 2: Setup IP address and directories: MODIFY THESE!!!!!!!!!!!!!!!!!!!
%=========================================================================

%IP address of camera
IP='192.168.2.155';

%camera name(choose something relatively short)
Cam='PuuOo';

%name the time zone used for this computer's clock (i.e. image time stamp)
timezone='HST';

%get watermark
cd('C:\Mikron');
wm=imread('usgslogo_white.jpg');  %this is just a USGS logo overlay on image

%cd to camera image directory (archive subfolders will be created here)
cc='C:\Mikron\PuuOo';
cd(cc);

%=========================================================================
%Part 3: Connect to camera and acquire single image
%=========================================================================

%ping camera 10 times before connecting
system(['ping ' IP ' -n 10']);

%run .exe to acquire Mikron binary image
dos('C:\***PutPathHere***\MikronForm05.exe');

%load and process binary image
cd(cc)
try
```

```
    fid=fopen('CurrentImage.bin','r','ieee-le'); %create binary image file name
    A=fread(fid,[320,240],'int16','ieee-le'); %read in image to Matlab array
    A=double(A);
    A=A./10;  %convert to Kelvin
    A=A-273.15; %convert to Celsius
    A=rot90(A,-1); %orient image correctly
    A=fliplr(A);
    fclose(fid)
catch exception
    dos('taskkill /F /IM MikronForm05.exe'); %kill function if failure
    dos('taskkill /F /IM cmd.exe');
    exit
end

d=dir('*.bin');
yy=datestr(d(end).date,10); %year
m=datestr(d(end).date,5); %month
dd=datestr(d(end).date,7); %day
hh=datestr(d(end).date,'HH'); %hour
mm=datestr(d(end).date,'MM'); %minute
ss=datestr(d(end).date,'SS'); %second

%=========================================================================
%Part 4: Rename and archive .bin file in correct folder
%=========================================================================
t=now; %current date/time
%rename .bin file to date-based name
d=dir('*.bin');
for i=1:length(d)
    year=datestr(d(i).date,10);
    month=datestr(d(i).date,5);
    day=datestr(d(i).date,7);
    hour=datestr(d(i).date,'HH');
    mint=datestr(d(i).date,'MM');
    sec=datestr(d(i).date,'SS');
    Aname2=[year month day hour mint sec 'M.bin']; %filename based on time
    if strcmp(d(i).name,Aname2)==0
        movefile(d(i).name,Aname2); %change name
    end
end

%move .bin file to right directory
minute=mint;

d1=dir('*.bin');
for x=1:length(d1)
    Aname2=d1(x).name; %get filename
    year=datestr(d1(x).date,10);
    month=datestr(d1(x).date,5);
    day=datestr(d1(x).date,7);
    hour=datestr(d1(x).date,'HH');
    mint=datestr(d1(x).date,'MM');
    sec=datestr(d1(x).date,'SS');

    cd('D:\Mikron\PuuOo\');
    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);
%uncomment line below if using hour folders, and comment out line after that
movefile(Aname2,['D:\Mikron\PuuOo\' year '\' month '\' day '\' hour ...
    '\' Aname2]);
%movefile(Aname2,[year '/' month '/' day '/' Aname2]);

cd('D:\Mikron\PuuOo\');
cd(year)
```

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

    Acut=A(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]);
    if maxA<10
        exit
    end
    imagesc(A,[10 0.95*maxA]); %display Matlab array as image
    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 colorbar is optional
    colormap(iron)
    %timestamp:
    text(10,10,['PTcam: ' m '-' dd '-' yy ' ' hh ':' mm ':' ss ' HST'],...
        'fontsize',10,'BackgroundColor','white','Margin',2);
    print('-r150','-djpeg85',[year month day hour mint sec 'M.jpg']); %make jpg image
    try
        %copy image to webserver computer:
        dos(['robocopy D:\Mikron\PuuOo\' year '\' month '\' day '\' hour ...
            '\ L:\cams\PTcam\images\archive\' year '\' month '\' day '\' ...
            hour '\ /COPY:DT /R:1 /E /XF Thumbs.db']);
    catch exception
        exception
    end
    cd(cc)
end


%=========================================================================
%Part 6: If the binary image was acquired correctly then do analysis and
%put the image in the right directory (if not, just give up and exit)
%=========================================================================

if size(A,1)>3
    cd(cc)
    %=====================================================================
    %Part 6a: Make simple temperature measurements
    %=====================================================================

    %measure maximum temperature in image
    Tmax=max(max(A))

    %measure temperatures at three spots specified in Part 1
    T1=A(r1,c1)
    T2=A(r2,c2)
    T3=A(r3,c3)

else exit
end
```

```
%=========================================================================
%Part 7: export image to jpg for easy html viewing
%=========================================================================
% setup figure window
h=figure(1);
set(h,'Units','inches')
set(h,'Position',[0 0 8 6])
set(h,'PaperPosition',[0 0 8 6])

Acut=A(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(A,[10 0.95*maxA]); %display array as image
hold on
image(wm,'Xdata',[2 27],'Ydata',[234 239]); %overlay USGS logo
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 %use iron colormap (optional)
colormap(iron)
%timestamp:
text(10,10,['PTcam: ' m '-' dd '-' yy ' ' hh ':' mm ':' ss ' HST'],...
    'fontsize',10,'BackgroundColor','white','Margin',2);

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

try
    cd('D:\Mikron\PuuOo\webserver');
    print('-r150','-djpeg85','M.jpg');
    print('-r45','-djpeg85','M.thumb.jpg');
    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 D:\Mikron\PuuOo\webserver L:\cams\PTcam\images /COPY:DT /R:1');
catch exception
end

%=========================================================================
%Part 8: write temperature measurements to a text file for simple analysis
%=========================================================================
cd(cc)
fid=fopen([Cam '_TempData.csv'],'at+');
k=273.15;
k=0;
fileline=[datestr(t,0) ',' num2str(Tmax-k) ',' num2str(T1-k) ',' ...
    num2str(T2-k) ',' num2str(T3-k)];
count=fprintf(fid,'%s\n',fileline);
st=fclose(fid);
```

```
%=========================================================================
%Part 9: delete incorrectly acquired binary images
%=========================================================================
cd(cc)
d=dir('*.bin');
for i=1:length(d)
    if d(i).bytes<20
        delete(d(i).name);
    end
end

exit %exit Matlab upon completion
```

*Script L1b: Connect to camera and acquire binary image*

The script below forms the main code for the executable MikronForm05.exe and was written in Visual Basic .NET, using Microsoft Visual Studio Express 2010. It accesses the ActiveX commands in the Mikron SDK (Software Developer's Kit) to acquire a single image from the Mikron M7500 thermal camera. This script is run by Matlab script L1, which is run periodically by Windows Scheduler. The Mikron ActiveX control must already be installed on the computer (this is installed automatically when installing Mikrospec RT_Multi). Script was written by Matt Patrick with substantial help from Kreg Kelley (Lumasense).

```
Imports System.Runtime.InteropServices
Imports AxMIKRONCAMERA7500CLIENTCONTROLLib
Imports System.Console
Imports System.IO
Imports System.IO.BinaryWriter

Public Class Form1

    Dim M7500Control As
AxMIKRONCAMERA7500CLIENTCONTROLLib.AxMikronCamera7500ClientControl
    Dim ptrArray As IntPtr
    Dim ImageArray() As Short

    Dim ArraySize As Integer = 76800
    Dim ArrayBytes As Integer = 153600

    Private Sub AcquireOne(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick

        Timer1.Stop()

        M7500Control = New
AxMIKRONCAMERA7500CLIENTCONTROLLib.AxMikronCamera7500ClientControl

        M7500Control.CreateControl()

        M7500Control.CameraNumber = 3

        Dim y As Integer
        y = M7500Control.GetNumberCameras()

        Dim bResult As Boolean

        bResult = M7500Control.Initialize()

        If bResult = True Then
            ReDim ImageArray(0 To ArraySize - 1)
            ptrArray = Marshal.AllocHGlobal(ArrayBytes)
        End If

        bResult = M7500Control.SetLens(2)

        bResult = M7500Control.SetCurrentCameraAndRange(1, 0)
```

```
        bResult = M7500Control.AcquiretoLVImage(ptrArray)

        M7500Control.Uninit()

        If bResult = True Then
            Marshal.Copy(ptrArray, ImageArray, 0, ArraySize)

            Dim i As Integer
            Dim c As String = ","
            Dim s As String = CStr(ImageArray(0))
            Dim st As String

            Dim binwriter = New BinaryWriter( _
                    File.Open("C:\Mikron\MT\CurrentImage.bin", FileMode.Create))

            For i = 1 To ArraySize - 1
                binwriter.Write(ImageArray(i))
            Next
            binwriter.Close()

        End If

        If ptrArray <> Nothing Then
            Marshal.FreeHGlobal(ptrArray)
        End If

        Erase ImageArray
        Application.Exit()
    End Sub

End Class
```

*Script L2: Track temperature trends*

```
function temperaturetrends2
%This Matlab function analyzes an incoming stream of thermal images and
%reduces the images to simple trends in temperatures to compare with other
%time series data. The mean and max of the images is recorded, as well as
%the percentage of the image which is above some threshold (e.g. 50 C, 100
%C, etc.).  The script then takes these resutls and writes them to a text
%file that is formatted for ingestion into VALVE.
%
% Matthew R Patrick
% Hawaiian Volcano Observatory
% US Geological Survey
%
% July 4, 2013
%

%go to where Matlab .mat files are
cd('C:\Documents and Settings\IGSWHVWG-geod\My Documents\MATLAB');

load temperaturelasttime %stores time of last image analyzed

stoptime=now-(10/1440); %only go up to 10 minutes ago

%ETcam initialization
ETtimes=[];
ETmean=[];
ETmax=[];
ET50=[];
ET100=[];
ET200=[];
ET300=[];
ET400=[];
%WTcam initialization
WTtimes=[];
WTmean=[];
WTmax=[];
WT50=[];
WT100=[];
WT200=[];
WT300=[];
WT400=[];
%PTcam initialization
PTtimes=[];
PTmean=[];
PTmax=[];
PT50=[];
PT100=[];
PT200=[];
PT300=[];
PT400=[];

numpix=320*240;
%ETcam:
%image without sky: A(82:230,10:310);
%WTcam:
%image without sky: A(94:230,10:310);

%for each camera, do this analysis
for jj=[1 3] %do these cameras
    Amax=[];
```

```
Amean=[];
A50=[];
A100=[];
A200=[];
A300=[];
A400=[];
thistimes=[];
key=0;
if jj==1
    cd('D:\Mikron\ET');
    lasttime=ETlasttime;
    stoptime=now-(7/1440);
elseif jj==2
    cd('D:\Mikron\WT');
    lasttime=WTlasttime;
    stoptime=now-(7/1440);
elseif jj==3
    cd('D:\Mikron\PuuOo');
    lasttime=PTlasttime;
    stoptime=now-(3.5/1440)
end

if lasttime>now
    lasttime=now-(2/24);
end

lastyear=datestr(lasttime,10);
lastmonth=datestr(lasttime,5);
lastday=datestr(lasttime,7);
lasthour=datestr(lasttime,'HH');
lastminute=datestr(lasttime,'MM');
lastsecond=datestr(lasttime,'SS');

%go through folders and find the last image that was analyzed.  This
%has the time variable "lasttime", which confusingly becomes
% the start time of this analysis.
dyear=dir;
for i=1:length(dyear)
    if strcmp(dyear(i).name,lastyear)==1
        dy=i;
        cd(lastyear)
    end
end
dmonth=dir;
for i=1:length(dmonth)
    if strcmp(dmonth(i).name,lastmonth)==1
        dm=i;
        cd(lastmonth)
    end
end
dday=dir;
for i=1:length(dday)
    if strcmp(dday(i).name,lastday)==1
        dd=i;
        cd(lastday)

    end
end
dhour=dir;
for i=1:length(dhour)
    if strcmp(dhour(i).name,lasthour)==1
        dh=i;
        cd(lasthour)
```

```
    end
end
dimages=dir('*bin');
di=1;
for i=1:length(dimages)
    s=dimages(i).name;
    currentyear=str2num(s(1:4));
    currentmonth=str2num(s(5:6));
    currentday=str2num(s(7:8));
    currenthour=str2num(s(9:10));
    currentminute=str2num(s(11:12));
    currentsecond=str2num(s(13:14));
    currenttime=datenum(currentyear,currentmonth,currentday,...
        currenthour,currentminute,currentsecond);
    if currenttime==lasttime
        di=i;
    end
end

count=0;
s=dimages(di).name;
thisyear=str2num(s(1:4));
thismonth=str2num(s(5:6));
thisday=str2num(s(7:8));
thishour=str2num(s(9:10));
thisminute=str2num(s(11:12));
thissecond=str2num(s(13:14));
thistime=datenum(thisyear,thismonth,thisday,thishour,thisminute,thissecond);

%now go through the images that have not yet been analyzed, and stop at
%"stoptime"
while thistime<stoptime
        if key==0
            key=1;
            di=di+1;
        end
        if di<=length(dimages)
            %import image
            FID=fopen(dimages(di).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

            %analyze
            if jj==1 %ETcam
                A=A(82:235,2:318); %cut out sky for ETcam
            elseif jj==2 %WTcam
                A=A(94:235,2:318); %cut out sky for WTcam
            elseif jj==3 %PTcam
                A=A(76:235,2:318); %cut out sky for PTcam
            end

            s=dimages(di).name;
            thisyear=str2num(s(1:4));
            thismonth=str2num(s(5:6));
            thisday=str2num(s(7:8));
            thishour=str2num(s(9:10));
            thisminute=str2num(s(11:12));
            thissecond=str2num(s(13:14));
            thistime=datenum(thisyear,thismonth,thisday,thishour,...
```

```
        thisminute,thissecond);

    thistimes=[thistimes thistime];

    %calculate parameters for VALVE
    Amean=[Amean mean(A(:))];
    Amax=[Amax max(max(A))];
    A50=[A50 (sum(A(:)>50)/numpix)*100];
    A100=[A100 (sum(A(:)>100)/numpix)*100];
    A200=[A200 (sum(A(:)>200)/numpix)*100];
    A300=[A300 (sum(A(:)>300)/numpix)*100];
    A400=[A400 (sum(A(:)>400)/numpix)*100];

    %continue with navigation, going through folders and images
    count=count+1;
    di=di+1;
elseif di>length(dimages) %if at end of hour folder
    cd ..
    dh=dh+1;
    if dh<=length(dhour)
        cd(dhour(dh).name)

        dimages=dir('*bin');
        di=1;
        length(dimages);
    elseif dh>length(dhour)
        cd ..
        dd=dd+1;
        if dd<=length(dday)
            cd(dday(dd).name);
            dh=3;
            dhour=dir;
            cd(dhour(dh).name)
            dimages=dir('*bin');
            di=1;
        elseif dd>length(dday)
            cd ..
            dm=dm+1;
            if dm<=length(dmonth)
                cd(dmonth(dm).name);
                dd=3;
                dday=dir;
                cd(dday(dd).name);
                dh=3;
                dhour=dir;
                cd(dhour(dh).name);
                dimages=dir('*bin');
                di=1;
            elseif dm>length(dmonth)
                cd ..
                dy=dy+1;
                cd(dyear(dy).name);
                dm=3;
                dmonth=dir;
                cd(dmonth(dm).name);
                dd=3;
                dday=dir;
                cd(dday(dd).name);
                dh=3;
                dhour=dir;
                cd(dhour(dh).name);
                dimages=dir('*bin');
                di=1;
```

```
                    end
                end
            end
        end

        if jj==1 %ETcam
            ETtimes=datestr(thistimes+10/24,31);
            ETmean=Amean';
            ETmax=Amax';
            ET50=A50';
            ET100=A100';
            ET200=A200';
            ET300=A300';
            ET400=A400';
            if length(thistimes)>0
            ETlasttime=thistimes(end);
            end
        elseif jj==2 %WTcam
            WTtimes=datestr(thistimes+10/24,31);
            WTmean=Amean';
            WTmax=Amax';
            WT50=A50';
            WT100=A100';
            WT200=A200';
            WT300=A300';
            WT400=A400';
            WTtimes
            if length(thistimes)>0
            WTlasttime=thistimes(end);
            end
        elseif jj==3 %PTcam
            PTtimes=datestr(thistimes+10/24,31);
            PTmean=Amean';
            PTmax=Amax';
            PT50=A50';
            PT100=A100';
            PT200=A200';
            PT300=A300';
            PT400=A400';
            if length(thistimes)>0
            PTlasttime=thistimes(end);
            end
        end
    end %end while loop
end %end for loop of cameras

cd('C:\Documents and Settings\IGSWHVWG-geod\My Documents\MATLAB');
save temperaturelasttime ETlasttime WTlasttime PTlasttime

%create text file of results for VALVE
cd('D:\Mikron\valvefiles');
fid=fopen([datestr(now,'yyyymmddHHMM') 'temptrends.txt'],'w');

for i=1:size(ETmax,1)
    fileline1=[ETtimes(i,:) ', ETcam, ' num2str(ETmean(i)) ', ' ...
        num2str(ETmax(i)) ', ' num2str(ET50(i)) ', ' num2str(ET100(i)) ...
        ', ' num2str(ET200(i)) ', ' num2str(ET300(i)) ', ' num2str(ET400(i))];
    count=fprintf(fid, '%s\n', fileline1);
end
for i=1:size(WTmax,1)
    fileline1=[WTtimes(i,:) ', WTcam, ' num2str(WTmean(i)) ', ' ...
        num2str(WTmax(i)) ', ' num2str(WT50(i)) ', ' num2str(WT100(i)) ...
        ', ' num2str(WT200(i)) ', ' num2str(WT300(i)) ', ' num2str(WT400(i))];
```

```
    count=fprintf(fid, '%s\n', fileline1);
end
for i=1:size(PTmax,1)
    fileline1=[PTtimes(i,:) ', PTcam, ' num2str(PTmean(i)) ', ' ...
        num2str(PTmax(i)) ', ' num2str(PT50(i)) ', ' num2str(PT100(i)) ...
        ', ' num2str(PT200(i)) ', ' num2str(PT300(i)) ', ' num2str(PT400(i))];
    count=fprintf(fid, '%s\n', fileline1);
end
st=fclose(fid);

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

copyfile(s1,s2); %copy text file of data to webserver computer for VALVE

exit %exit Matlab
```

*Script L3: Alarm if temperature anomaly detected*

```
function temperaturealarmETWT
%Matlab function looks at most recent image and checks for anomalously high
%temperatures. If high temperature is found, it sends a text message.
%
% Matthew R Patrick
% US Geological Survey
% Hawaiian Volcano Observatory
%
% June 26, 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.

cd('D:\Mikron');

load tempalarmtime %load time of last alert

cd('D:\Mikron\ET');

year=datestr(now,'yyyy');
cd(year) %go to current year folder
d1=dir;
cd(d1(end).name); %go to most recent month folder
d2=dir;
cd(d2(end).name); %go to most recent day folder
d3=dir;
cd(d3(end).name); %go to most recent hour folder
d4=dir('*bin');

FID=fopen(d4(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
%cuts out sky, where sun is apt to cause false alarm (modified 8/26/11)
Acut=A(82:235,2:318);
maxA=max(Acut(:));
thresh=120;  %temperature threshold in Celsius

ETtimediff=now-lastETtime; %time since last alert

%if temperature over threshold and last alert was more than 1 day ago
if maxA>thresh & ETtimediff>1
    s=['Auto alert: Max temp. in ETcam is ' num2str(maxA) ...
        ' There may be lava!'];
    try
        send_text_message('xxx-xxx-xxxx','verizon','high temp',s)
    catch exception
    end
    lastETtime=now;
end
```

```
cd('D:\Mikron');
save tempalarmtime lastETtime %save last alert time

exit %exit Matlab
```

*Script L4: Kill Matlab*

This very simple batch file holds a single DOS command and is run periodically by Windows Scheduler to kill Matlab, to ensure that multiple instances of Matlab do not accumulate and freeze the acquisition computer.

```
taskkill /F /IM MATLAB.exe
```

*Script L5: Make composite image*

```
function mikroncomposite3
%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
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 max temperature array
dates=[];
maxtemp=[];
for d1=1:2 %go through previous day and current day
    if d1==1
        cd(['D:/Mikron/PuuOo/' startyear '/' startmonth '/' startday])
    elseif d1==2
        cd(['D:/Mikron/PuuOo/' 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
            hrs(i).name
            pwd
            cd(hrs(i).name);
            d=dir('*.bin'); %list all images in this frame
            for j=1:length(d) %go through all images in this frame
                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=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
            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); %set up 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(:)) %min temp for colorscale
maxA=max(Acut(:)) %max temp for colorscale
ah=axes('Position',[0 0 1 1]);
imagesc(Amax,[10 0.95*maxA]); %display array as image
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 works too
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('D:\Mikron\PuuOo\composites');
    print('-r150','-djpeg85','M.jpg'); %write jpg file
    fid=fopen('js.js','wt'); %make accessory file
    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 D:\Mikron\PuuOo\composites L:\cams\PTcam\composites /COPY:DT /R:5');
catch exception
end

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

cc='D:\Mikron\PuuOo\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',['PTcam' year month day 'M.jpg']); %make jpg image
%movefile(Aname2,[year '/' month '/' day '/' hour '/' Aname2]);
%movefile(Aname2,[year '/' month '/' day '/' Aname2]);

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

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

exit %exit Matlab
```

*Script L6: Cycle power on camera*

```
function pulsePTcam
%Matlab function cycles the power on a thermal camera using a power relay web
%address.  Function is designed to be run using Windows Scheduler once a
%week.
%
% Matthew R Patrick
% US Geological Survey
% Hawaiian Volcano Observatory
%
% June 26, 2013
%

%new power relay on puuoo installed sept 2012
system('wget http://192.168.190.78/state.xml?relay1state=2&pulseTime1=5');

%set up email preferences and send email update
setpref('Internet','SMTP_Server','xxxxx.xxxx.usgs.gov');
setpref('Internet','E_mail','xxxxxxxxx@usgs.gov');
sendmail('xxxxxx@gmail.com','Pulse PTcam','it worked');

exit %exit Matlab
```

*Script L7: Check acquisition*

```
function checkacquisition
%This Matlab function checks to make sure images are still incoming, and if not,
%sends an email alert
%
% Matthew R. Patrick
% US Geological Survey
% Hawaiian Volcano Observatory
%
%
% June 25, 2013
%

%cd to camera directory, which holds date-based folders
cd('D:\Mikron\PuuOo');

year=datestr(now,'yyyy');
cd(year) %cd into current year folder
d1=dir;
cd(d1(end).name); %cd into last month folder
d2=dir;
cd(d2(end).name); %cd into last day folder
d3=dir;
cd(d3(end).name); %cd into last hour folder
d4=dir('*bin');

timediff=now-d4(end).datenum; %time since last image, in day fraction
timediffminutes=timediff*1440; %convert to minutes

%if last image is older than 10 minutes old, then send email
if timediffminutes>10
    %then email me at my gmail account
    setpref('Internet','SMTP_Server','xxxx.xxxx.usgs.gov');
    setpref('Internet','E_mail','xxxxxxx@usgs.gov');
    sendmail('xxxxxxxx@gmail.com','Acquisition stall on PTcam',['Last image: '
d4(end).date]);
end

exit %exit Matlab
```