



"HENRI COANDA"
AIR FORCE ACADEMY
ROMANIA



"GENERAL M.R. STEFANIK"
ARMED FORCES ACADEMY
SLOVAK REPUBLIC

INTERNATIONAL CONFERENCE of SCIENTIFIC PAPER
AFASES 2013
Brasov, 23-25 May 2013

MEANS OF EXTENDING VISION FROM VISIBLE TO INFRARED SPECTRUM AND PRACTICAL IMPLEMENTATION

Constantin STRIMBU*, Eduard LUCHIAN**

*Academia Fortelor Aeriene "Henri Coanda", Brasov, Romania, **Facultatea de Electronica, Telecomunicatii si Tehnologia Informatiei, Universitatea Tehnica, Cluj-Napoca, Romania

Abstract: This paper intends to make a brief introduction to the principles of infrared light exploring at the same time capabilities of harnessing the new technologies. The paper approaches the design, explanation and construction of an infrared vision capable webcam. Several comparisons are made though out the paper highlighting the benefits that the proposed design brings. Finally after a MATLAB and a XARA process a possible working design is offered.

Keywords: infrared, pixel, capture, spectrum, webcam.

1. INTRODUCTION

In order to understand infrared vision, it is important to understand something about light. The amount of energy in a light wave is related to its wavelength: Shorter wavelengths have higher energy. Of visible light, violet has the most energy, and red has the least. Just next to the visible light spectrum is the **infrared** spectrum [1].

Infrared light can be split into three categories:

- **Near-infrared** (near-IR) - Closest to visible light, near-IR has wavelengths that range from 0.7 to 1.3 **microns**, or 700 billionths to 1,300 billionths of a meter.
- **Mid-infrared** (mid-IR) - Mid-IR has wavelengths ranging from 1.3 to 3 microns. Both near-IR and mid-IR are used by a variety of electronic devices, including remote controls.
- **Thermal-infrared** (thermal-IR) - Occupying the largest part of the infrared spectrum, thermal-IR has wavelengths ranging from 3 microns to over 30 microns.

The key difference between thermal-IR and the other two is that thermal-IR is **emitted** by an object instead of reflected off it. Infrared light is emitted by an object because of what is happening at the **atomic** level.

2. PROJECT DESIGN

In the project there were used two web cameras in order to capture images. One was left unmodified (UW) and the other (MW) (see figure 1). An visible light filter and a set of 10 LED's with infrared light were attached in order to be able to see at night.



Figure 1. Modified Webcam

The circuit had to be adapted to 5 volts which is the USB output. In order to have a functional configuration, each LED was stringed with a 68 Ω resistance that makes the

circuit functional. The circuit's electrical design is presented in figure 2.

What this system brings new is the fact that different risk factors can be stored in a memory and whenever the system has a live streaming it can automatically compare the image with the risk factors that were stored and decide the course of action which at first would be signaling the presence of a potentially dangerous factor in the surrounding's.

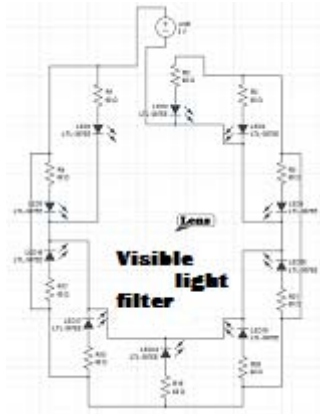


Figure 2. Electrical design.

In order to prove the efficiency of such a system the following experiment was carried out:

Captured images with the two webcams were used to highlighting the advantages that the modifications brought.

In the images captured with the UW we can see the light but can't identify the source or the objects near it. In the pictures with the MW we can clearly see what is next to the flame of the candle. See figure 3 and 4.

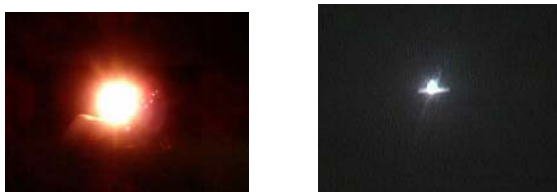


Figure 3. UW and MW image

This principle can be applied to any devices that captures images with light emitting sources.



Figure 4. UW and MW image

Here is an example of night vision of this system in which with the UW nothing can be seen but the MW can easily see the infrared filter leaving the full image to be seen (see figure 5).



Figure 5. UW and MW image

3. MATLAB PROCESSING

In the image captured with the MW you can spot a human (figure 6) form in night vision. This image was processed [2] in MATLAB and the form was added to a database in order to be recognized. This can be done based on recognizing the group of pixels that make up whatever it's in the database. If we add a variable number of pixels [5] to any image we can enlarge the range that system recognizes a form.

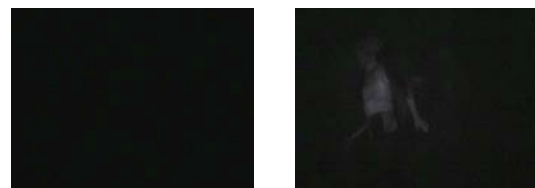


Figure 6. UW and MW image

An image can be displayed (in MATLAB) as a matrix starting from the following representation for a digitized image function (1).

$$f(x,y) = \begin{bmatrix} f(0,0) & \dots & f(0,N-1) \\ f(1,0) & \dots & f(1,N-1) \\ \dots & \dots & \dots \\ f(M-1,0) & \dots & f(M-1,N-1) \end{bmatrix} \quad (1)$$

The right side of this equation is a digital image by definition each element of this array called image element, picture element, pixel or



"HENRI COANDA"
AIR FORCE ACADEMY
ROMANIA



"GENERAL M.R. STEFANIK"
ARMED FORCES ACADEMY
SLOVAK REPUBLIC

INTERNATIONAL CONFERENCE of SCIENTIFIC PAPER
AFASES 2013
Brasov, 23-25 May 2013

peel. The terms image a pixel will be used in the following discussion to denote the image and its elements.

A digital image can be natural represented as a MATLAB matrix [3] (2).

$$f(x,y) = \begin{bmatrix} f(1,1) & \dots & f(1,N) \\ f(2,1) & \dots & f(2,N) \\ \dots & \dots & \dots \\ f(M,1) & \dots & f(M,N) \end{bmatrix} \quad (2)$$

Displaying the image you have just read is achieved by simply typing: **image(x)**

This will open a Figure Window and plot the image on the axis. If you have read the JPEG image above, **image** will readily accept the 3-D array and display the RGB images. Notice that the same plotting considerations of relative axis scale arise here as well; you might wish to use **axis equal** to correct the perspective. If the image you read is an indexed image, i.e., a 2-D index array with a corresponding Nx3 color map, you will need to use the **color map** function to get the correct coloring of the plot. At the website you can download the indexed image *usflag.dib* (Windows device-independent bitmap).

Once you have that file you can read it, then view it with:

```
[x,cm]=imread('usflag.dib','bmp');
image(x)
```

Writing the contents of a Figure Window to an image file is just as simple as reading one. The **imwrite** function[4] provides a means to create image files. The general form of **imwrite** is:

```
imwrite(A,filename,fmt)
```

where *A* is the image, either grayscale if NxM, or truecolor if NxMx3,

filename is a string containing the name of the file to be created, and *fmt* is a

string indicating one of the write formats indicated in MATLAB documentary.

For the case of an indexed image, i.e., one containing an image and colormap, **imwrite** takes the following form:

```
imwrite(X,C,filename,fmt)
```

By indexed we mean that the image is created from information in two arrays: the first is an array of indexes into the second, which is a three-column array containing the red, green, and blue contributions for each pixel. The following code will load an image of a clown and display it in a Figure

Window, looking like that shown in figure 7.

```
load shape
image(X)
colormap(map)
```



Figure 7. Image window

When we load *image.mat* the arrays *X* and *map* were placed in the MATLAB workspace. The array *X* is an array of indices; row indexes actually, specifying which row from *map* (an 81x3 array) the pixel color is to be taken from.[3]

Looking at a few pixels from the upper left corner, we see in figure 8.

Each element is a number corresponding to a row in the color map. We can find the color

(figure 9) components for each of these pixels by:

```
- Map (unique(X(60:40,60:40)),:)
```

```
ans =
```

```

     2     2     2     2     2
    61    69    69    69    69
    69    61    69    61    69
    61    69    61    61    56
    69    55    61    44    61

```

Figure 8. Pixels

```
ans =
```

```

 0.1250         0         0
 0.8672    0.4141    0.1250
 0.8672    0.5781         0
 0.8672    0.5781    0.1250
 0.9961    0.5781    0.1250
 0.9961    0.7031    0.1250

```

Figure 9. Color code

The **unique** function returns only one instance of an element, and orders the results, so the first row here corresponds to an index value of 2, the second 44, the third 55, etc.. We can see these first few pixels, shown in figure 10, with the following code [4].

```
image(X(60:40,60:40))
colormap(map)
```



Figure 10. Graphic pixel display

Based on those functions all the pixels that are different from black were selected. And after a further process with XARA resulting in a database form (figure11):

This is an example of shape that can be stored in a system memory in order to compare live streaming with it.



Figure 13. Database form

3. CONCLUSIONS & ACKNOWLEDGMENT

After the experiment we can state that implementing such a design in modern surveillance systems could make a difference because they could be set to inform appropriate personal about the existing of a possible threat. Thus optimizing resources.

Applying an infrared filter to a camera refines the image that is captured making it more easily to see all the elements of the image excluding possible false threats and giving the image processing a better analyzing perspective.

The infrared LED's offers substantial extension to the capabilities of the camera and because they emit in infrared they can't be spotted and the surveillance camera is not unmasked.

REFERENCES

1. Jeff Tyson *How Night Vision Works* Available at: <http://electronics.howstuffworks.com/gadgets/high-tech-gadgets/nightvision1.htm> (January 2013)
2. Gonzalez, Woods & Eddins *Digital Image Processing Using Matlab* Pearson Prentice Hall (December 26, 2003)
3. Shehrzad, Qureshi *Embedded Image Processing with DSP Examples in MATLAB* Springer (April 7, 2005)
4. Patrick Marchand and O. Thomas Holland *Graphics and GUIs with MATLAB* Chapman and Hall/CRC; 3 edition (November 25, 2002)
5. Eduard CIUDATU *Tendințe și orientări în dezvoltarea sistemului de dirijare la racheta antiaeriană a-94*. AFAHC (July, 2011)



"HENRI COANDA"
AIR FORCE ACADEMY
ROMANIA



"GENERAL M.R. STEFANIK"
ARMED FORCES ACADEMY
SLOVAK REPUBLIC

INTERNATIONAL CONFERENCE of SCIENTIFIC PAPER
AFASES 2013
Brasov, 23-25 May 2013