



MATLAB, Mathematics, and Engineering (MAT-ME) Summer Camp for Preparing Students for Engineering Subjects

Reza Fazel-Rezai, Sima Noghianian, and Ahmed Rabbi

Department of Electrical Engineering, University of North Dakota

reza.fazel-rezai@engr.und.edu, sima.noghianian@engr.und.edu,

ahmed.rabbi@my.und.edu

Abstract

Professors in the engineering programs see many freshmen engineering students struggle with the intensive math contents of the courses. This is mainly because the materials are more difficult and different than the math they learned in high school. While engineering courses all need strong background in mathematics, the application of mathematics in engineering problems is much more important than the theory. It is impossible to fully master almost any topic in engineering without adequate skills in mathematics and programming. Therefore, having a summer camp that introduces the mathematics through programming and hand-on experience provides good introduction to engineering and provides a tool for recruiting students to engineering programs. It also helps student to expand their understanding of math to applications. This paper describes the series of lessons that we developed using MATLAB¹ (developed by Mathworks®) with the focus on signal and image processing at the University of North Dakota College of Engineering and Mines in Grand Forks, North Dakota. The summer camp was named MAT-ME (**MATLAB, Mathematics, and Engineering**) and was aimed to promote Math and Engineering among high school students in the state of North Dakota. We ran the one-week long MAT-ME camp three times in summer of years 2010, 2011, and 2012. In this paper, we will describe the teaching materials that we developed, the results of students and parents' evaluations, and the lessons learned during the three years that the camp was held.

Introduction

Well-documented trends have been reported nationally of declining interest, poor preparedness, a lack of diverse representation, and low persistence of U.S. students in STEM (Science, Technology, Engineering and Mathematics) disciplines. A national crisis has been identified in the area of global technological competitiveness². In engineering, the need for change has been highlighted by American Society for Engineering Education's (ASEE) Engineering Deans Council and the Corporate Roundtable (1994); the National Research Council (1995); the National Academy of Engineering (2002 and following); and the National Science Foundation².

Engineering students who fail to see a connection between rigorous mathematics classes and the application of math often lack academic motivation. The objective of the MAT-ME camp was to provide an opportunity for students to apply math theory to real-life engineering problems. On the other hand, MathWorks' MATLAB is a software that is used widely in engineering courses such as Numerical Analysis, Signal Processing, Modeling of Dynamical Systems, Control Systems, Digital Control, Adaptive and Nonlinear Control, Biomedical Engineering, Antenna Theory, Transmission Lines, Electromagnetics, Computer Vision, Digital Image Processing, Chemical Process Control, Communication Theory, Wireless Systems, etc. MATLAB is very user friendly, easy to learn, and very powerful for engineering and math applications. Many real-world applications in engineering can be modeled, simulated and demonstrated by MATLAB. Many engineering textbooks now include examples based on MATLAB. Therefore, this software was chosen as the primary programming language for the MAT-ME camp.

There are various engineering concepts with simple mathematics that are understandable by high school students and can be implemented and demonstrated easily using MATLAB. Among these concepts are image and sound processing which has been used in other camps as well^{3,4}. As a part of the NSF Discovery K-12 program at Drexel University, during the 2007-2008 school year various signal processing activities were designed for high school students^{3,4}. During the 2000-2001 pilot school year, fourteen schools from the Dallas, Houston, and San Antonio, TX, areas taught the INFINITY Project curriculum⁵. A few colleges, such as Georgia Tech, have changed the curriculum and start with a Digital Signal Processing (DSP) course as oppose to analogy circuit courses⁶.

The goal in MAT-ME summer camp was to introduce basic math and engineering concepts through MATLAB programming and hands-on experiences to high school students. This camp was meant to be a good introduction to engineering and a tool for recruiting students to the engineering programs. Activities such as MAT-ME can help students to expand their understanding of math to its applications. Such a program has been introduced in other states such as a program at Georgia Institute of Technology and the Infinity Project sponsored by different US universities and high schools. This camp was a small step towards establishing a set of courses promoting math and engineering among high school students in the state of North Dakota.

The objectives of the MAT-ME camp were to:

- a) Expose senior high school students to math and programming with application in engineering through hands-on real-world projects,
- b) Increase motivation and practice deep thinking using problem-based learning methods,
- c) Teach them a powerful programming language (MATLAB) that they can use it for many different applications,
- d) Recruit students into University of North Dakota engineering program, and
- e) Provide fun and exciting experience of how math can be used for engineering problem solving.

Learning outcome of the camp can be summarized as:

1. Learn basic scientific programming skills and gain experience using MATLAB,
2. Gain confidence in applying math skills in solving real-life problems,
3. Learn problem solving skills, and
4. Learn how to work effectively in a team.

MAT-ME Camp Organization

This camp was organized and managed by Dr. Sima Noghianian and Dr. Reza Fazel-Rezai both Assistant Professors in the Department of Electrical Engineering, University of North Dakota (UND). They have more than 7 years of experience in teaching electrical engineering courses and more than 15 years experience using MATLAB. Since they joined UND in 2008, they have organized and participated in several other outreach programs such as participation in the Young Scientist and Engineer Academy (YSEA) camp, Dakota Science Center’s Supper Science Saturday (SSS). The camp was held in a computer laboratory where students could apply methods to the data gathered in lab.

The MAE-ME camp was designed for 5 full days of activities. The target audience was junior and senior high school students. However, it was open to freshman university students as well. Table I shows the major topics that were covered in each day of the camp.

Table I. The topics covered in each day of the camp

Day		Topics
Morning Schedule		Afternoon Schedule
Day 1	Introduction to MATLAB	<ul style="list-style-type: none"> • Building Simple Circuits • Problems of the Day
Day 2	Introduction to Image and Sound Signal Processing	<ul style="list-style-type: none"> • Sound, Frequency and Spectrum • Problems of the Day
Day 3	Experiment 1: Process Your Own Voice	<ul style="list-style-type: none"> • MindFlex Concentration Game • Problems of the Day
Day 4	Experiment 2: Process Your Own Image	<ul style="list-style-type: none"> • Reaction-Time Competition • Problems of the Day
Day 5	Competition	<ul style="list-style-type: none"> • Solution to the Competition Problem • Award Ceremony

Morning Activities

During Day 1 of the camp, participants became familiar with basics of MATLAB. They were taught basic programming skills in MATLAB through programs and games provided by MathWorks during morning classes (Fig. 1). The basic math they needed such as vector and matrices and vector algebra was reviewed and simple vector operator such as translating and rotating points of a figure were discussed. During Day 2

participants learned about the math skills they needed to work on sound and image signals for removing noise, filtering and processing.



Fig. 1. Morning computer class

During Day 3 and Day 4, some activities were done in the team format. Teams worked on the experiments designed to help them apply their knowledge of MATLAB and math and what they learned during Day 1 and Day 2 in processing their own voice (Day 3) and image (Day 4). Day 5 was the evaluation day. A challenging problem was given to participants to work on in the morning session and demonstrate their solution in the afternoon session.

Afternoon Activities

A. Hands-on Experiences

Each afternoon session was filled with hands-on activities and competitions. One of the activities was to build small projects and electric circuits using Electronic Snap Circuit Kits (<http://www.snapcircuits.net>). The kit comes with all the circuit parts and components in a shape that can be snapped to the circuit board (Fig. 2).



Fig. 2. Snap Circuit Kit, www.snapcircuits.net

During the second day, through a series of simple experiences students learned about signal and frequency (Fig. 3). They worked with function generator and oscilloscope to see different signal shapes and then they used microphones and speakers to see the shape of voice signal. They also worked with spectrum analyzer to experiment on how the change of signal shape changes the spectrum change.



Fig. 3. Afternoon hands-on activity

During Day 3, students practiced how to concentrate based on the MindFlex equipment (<http://mindflexgames.com>). This system uses brainwaves to calculate concentration of the subject. It is based on a wireless transmission of brainwaves using a simple headband in which there is only one dry electrode in contact with the subject's forehead. In this game, two participants can compete to see who can concentrate more. This was a good example of a device designed by engineers.

In the afternoon of Day 4, reaction time of the subjects were measured based on the BioPac equipment (<http://www.biopac.com>). The BioPac is a general purpose biomedical signal acquisition equipment with several lessons, one of experiments is to measure reaction time based on the difference time between hearing a sound and pressing a button. This was another fun competition for the participant to demonstrate what type of equipment can be designed by engineers.

B. Problems and Challenges of the Day

At the end of each day challenge problems were given to students to think about it and solve it as an assignment. The submitted solutions were contributing in the calculation of total rank for the final challenges. Samples of Problems of the Day are listed as follows:

Problem 1

Each letter in $SEND + MORE = MONEY$ (where SEND and MORE are 4-digit numbers, and MONEY is a 5-digit number) represents a positive distinct one-digit integer, with S and M are not zero. What does each letter represent?

Problem 2

Four people have one flashlight want to cross a bridge at night. A maximum of 2 people can cross at a time and every crossing needs a flashlight. One person takes 1 minute to cross, another takes 2 minutes to cross, another takes 5 minutes, and the last takes 10 minutes. If two people cross, it takes as long as the slower person takes (for example, the 1-minute person and 2-minute person cross in 2 minutes). It is possible to cross everyone in exactly 17 minutes. How?

Problem 3

Mr. and Mrs. Smith invited 5 other couples to a party. Each one of the 12 people at the party shook hands with some of the other people at the party, but no one shook hands with himself/herself, his/her spouse, or anyone twice. Mr. Smith asked the 11 other people at the party how times they shook hands, and everyone gave a different answer. How many times did Mr. Smith shake hands?

Problem 4

You are given 2 candles with non-uniform density and you know each burns in exactly one hour. How do you measure 45 minutes?

Problem 5

Place 5 black queens and 3 white kings on a 5 by 5 chessboard such that none of the kings are in check.

Problem 6

In $62 - 63 = 1$, move only one digit to make the statement true (the $-$ and $=$ are unmovable and switching two digits counts as 2 moves).

Problem 7

You are diagnosed with a disease that can only be cured by having one A pill and one B pill every morning for 15 consecutive days. Your doctor gives you a bag labeled "A" with 15 A pills inside and a bag labeled "B" with 15 B pills inside. Other than these, there are no more A or B pills in the world and you cannot make more. Also, A and B pills look the same and you cannot tell the difference by looking at them. So the first day, you take an A and B pill in the morning. The next morning, you take an A pill from its bag but accidentally take 2 B pills and mix it with the A. You still must have one A and one B pill for the next 14 days to be cured. How do you cure yourself?

Problem 8

Five people have one flashlight want to cross a bridge at night. A maximum of 2 people can cross at a time and every crossing needs a flashlight. One person takes 1 minute to cross, another takes 3 minutes to cross, another takes 8 minutes, another takes 10 minutes, and the last takes 17 minutes. If two people cross, it takes as long as the slower person takes (for example, the 1-minute person and 3-minute person cross in 3 minutes). It is possible to cross everyone in exactly 36 minutes. How?

Marketing Plan

The College of Engineering and Mines (CEM) graphic designer helped in designing advertisement to promote MAT-ME camp in the following places:

1. The Grand Forks Herald
2. UND's U-letter
3. CEM's K-12 Outreach website
4. Email-invitation to math teachers and school counselors in Red River and Central High Schools
5. Flyers distributed at various math competitions such as Math Kangaroo and UND Math Track Meet
6. Flyers during different events held by Dakota Science Center (DSC) such as and Super Science Day

MATLAB Lessons

As mentioned the focus of morning classes was to give students a breadth of knowledge in programming and in particular skills in MATLAB programming. Table II shows the details of lesson titles. In this paper, due to space limitations, we only focus on the sound and image processing parts of the lessons. Sample lessons are as follows.

Table II. Lesson titles

Topics
1. Introduction to MATLAB software
2. Defining variables, vectors, matrices in MATLAB
3. How to write MATLAB program
4. Examples of MATLAB programs
5. Manipulation of vectors and matrices in MATLAB
6. What is a sound and how can be processed in MATLAB
7. What is an image and how can be processed in MATLAB
8. Filtering signals in MATLAB
9. Recording your voice, loading, and plotting in MATLAB
10. Sound signal filters in MATLAB
11. Applying the filters to your voice
12. Write a MATLAB function to load, filter, and plot sound signals
13. Recording your image, loading, and displaying in MATLAB
14. Image filters in MATLAB
15. Applying the filters to your image
16. Write a MATLAB function to load, filter, and plot images; Print the results

Sound Signal Processing in MATLAB

By the time we started teaching this lesson on Day 3, students already knew how to open and write a program in MATLAB and how to show results in simple plots. The following

is the step by step lesson given as a manual to students:

Step 1: Clear your MATLAB work space and close all already open windows.

To do this you need to run the following simple commands in MATLAB:

```
clc % clear command line  
clear all % clear workspace  
close all % close all open window
```

Step 2: Record your voice using the microphone

You can do this in two different ways:

1. Using sound recording software:

First of all, verify whether your sound card is working properly. You can verify that your sound card is functioning properly by recording data and then playing back the recorded data. Recording data uses the sound card's analog input subsystem, while playing back data uses the sound card's analog output subsystem. Successful completion of these two tasks indicates your sound card works properly. The source of the recorded data is the microphone (built-in with webcam).

The first thing you should do is to enable your sound card's ability to record and play data. This is done using the Microsoft Windows Sounds and Audio Devices Properties dialog box. You can access this dialog box using the Windows **Start** button.

Select **Start > Settings > Control Panel**, then double-click **Sounds and Audio Devices**.

In the Sounds and Audio Devices Properties dialog box both playback and recording can be configured.

You can record data and then play it back using the Windows **Sound Recorder** panel. To access this application, select the following:

Start > Programs > Accessories > Entertainment > Sound Recorder

In case you don't find the Windows **Sound Recorder**, you should install a free sound recording software (for example, RecordPad Sound Recorder; you can download the rpsetup.exe from http://download.cnet.com/RecordPad-Sound-Recorder/3000-2170_4-10327499.html). Using this recorder you can record your own voice and save it in .WAV format.

If you have successfully recorded audio, skip to step # 5 and verify the .WAV file is saved on directory. Then come back at this point for recording audio directly in MATLAB.

2. Alternately, record directly in MATLAB.

```
% Find your MATLAB audio recording tool
S=which('wavrecord')
% Record 5 seconds of 16-bit audio sampled at 8000 Hz.
n=5; % 5 seconds; try it fun
Fs=8000; % 8000 samples/sec; try it fun
X = wavrecord(n*Fs,Fs,'double')
```

Step 3: Play the recording

Play the recorded audio by typing the following MATLAB commands:

```
SF=1 % speed factor: can be varied to play normal, fast forward
% default value = 1; try it for fun ...
wavplay(X,SF*Fs)
```

Step 4: Save the recording

Now you can save the recorded audio in Microsoft WAVE (.wav) sound file

```
% write wav file
wavwrite(X,'myrecording')
```

Step 5: Load the WAVE file in MATLAB and play it again

Load the .wav file that you just saved on hard disk drive. Make sure the WAV file is in directory.

```
[Z, Fs, nbits] = wavread('myrecording.wav');
```

Play again. You are welcome to play with varying value of “Fs” and see what happens.

```
wavplay(Z,Fs)
```

Step 6: Add some noise to the recorded audio

```
% Add Gaussian noise
x=Z;
x=x-mean(x); % remove DC bias
nSamples=length(x); % number of samples
se=norm(x,2)^2/nSamples; %signal energy
SNR=10; % signal to noise ratio
```

```

nsc=se/(10^(SNR/10)); % noise source
y=sqrt(nsc)*randn(nSamples,1);
ne=norm(y,2)^2/nSamples; %noise energy
fprintf('Estimated SNR=%f\n',10*log10(se/ne));
z=x+y; % noise corrupted signal
% Alternately, using MATLAB awgn function:
z=awgn(x,SNR); % noise corrupted signal

```

Step 7: Play the noise free (original) and noisy sound

```

Fs=8000; % 8000 samples/sec
wavplay(x,Fs)
wavplay(z,Fs)

```

Step 8: Plot the noise free and noisy sound signals

```

figure(1)
subplot(211),plot(x),title('Recorded sound')
subplot(212),plot(z),title('Noise corrupted recording')

```

Step 9: Filtering the noise and recover the original sound

```

% Filter the noise
% Mean or Average filtering
windowSize=3; % try it for fun
data_ave=filter(ones(windowSize,1),1,z);
% Median filtering
data_med=medfilt1(z>windowSize);

```

Step 10: Play and plot the filtered signals

```

% Play the filtered sounds
wavplay(data_ave,Fs)
wavplay(data_med,Fs)
% Plot again
figure(2)
subplot(211),plot(data_ave),title('After mean filtering')
subplot(212),plot(data_med),title('After median filtering')

```

Sample Results: Sample plotted results are shown in Fig. 4.

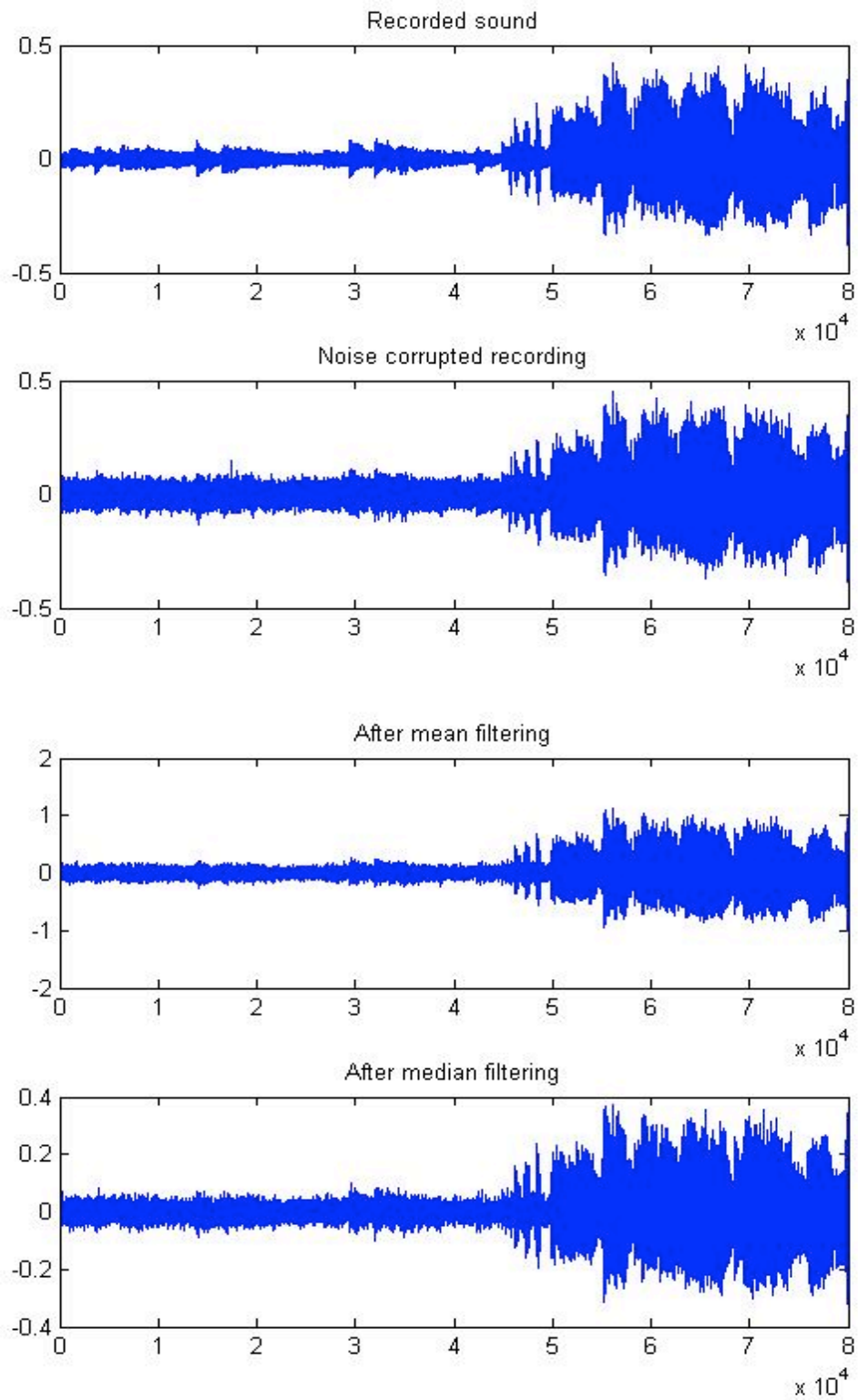


Fig. 4. Sample plots: recorded signal, after noise added, after mean filtering, and after median filtering.

Image Processing in MATLAB

Step 1: Capture image using webcam and print

Double click the webcam software (for example, Logitech QuickCam) icon on your desktop. Click on quick capture icon to take your picture using webcam. You can print your picture using the print icon on your webcam software taskbar. Alternately, find your picture in the location My Documents\My Pictures\ on your PC and print it.

Step 2: Scan the image

Scan your printed image using the scanner attached to your computer. In this way we can convert an analog image to a digital image. You can change the resolution (ppi – pixel per inch, for example, 50, 100, 150 and so on) of your image. Once you are done with scanning, save the scanned image to your MATLAB working directory.

Step 3: Load the image in MATLAB

These commands will help you to clear your MATLAB workspace memory and close all running or already open windows.

```
clc % clear command line
clear all % clear workspace
close all % close all open window
```

Now we are ready to load your image in MATLAB for processing it. We can use the following commands to find and load your image in MATLAB workspace from your computer's memory.

```
%Open standard dialog box for retrieving files
[Filename,Pathname]=uigetfile('*.*jpg')
A=imread(Filename) % Original RGB image
```

Step 4: Convert image from RGB to Gray scale for further processing.

```
B=rgb2gray(A); % Gray scale image
```

Step 5: Now let's have a look at the image. Note this one is the original color image.

```
% See your image
figure(1)
clf %Clear current figure window
colormap('gray'); % Set colormap to gray scale
subplot(221),imshow(A),title('Original image')
```

Note: MATLAB function `imagesc` also works the same way as `imshow`.

Step 6: Image Processing

At this stage, let's play with the image, add some noise and do some image processing. The command examples below will let to add two types of noises, Gaussian noise, and Salt and Pepper noise (on and off pixels)

```
C=imnoise(B,'gaussian'); % Gaussian noise
D=imnoise(B,'salt & pepper'); % Salt and pepper noise
```

Step 7: Now we will have a look at the noisy images.

```
% your image in gray scale
subplot(222),imagesc(B, [0 255]),title('Gray scale image')
% image with added Gaussian noise
subplot(223),imagesc(C, [0 255]),title('Gaussian noise')
% image with added salt and pepper noise
subplot(224),imagesc(D, [0 255]),title('Salt & pepper noise')
```

Step 8: Filtering the added noise and recover the original gray scale image.

In this step, we will perform two basic types of filtering on image. These are Mean or Average filtering which is usually used to remove Gaussian noise and Median filtering which is used to remove Salt and Pepper noise.

```
% Filtering the noise
win=3; % Two dimensional filter size. Play with the filter size for fun and see the results
% Mean or average filtering
Cout1=filter2(fspecial('average',win),C)/255;
Dout1=filter2(fspecial('average',win),D)/255;
% Median filtering
Cout2=medfilt2(C,[win win]);
Dout2=medfilt2(D,[win win]);
```

Step 9: Visually compare the results

```
% Compare the results
figure(2),
clf
colormap('gray');
% Mean filtering to image corrupted with Gaussian noise
subplot(221), imshow(Cout1),title('Mean filtering to Gaussian noise')
% subplot(221), imagesc(Cout1,[0 255]),title('Mean filtering to Gaussian noise')
% Median filtering to image corrupted with Salt and Pepper noise
subplot(222), imagesc(Cout2,[0 255]),title('Mean filtering to Salt&pepper noise')
```

```

% Mean filtering to image corrupted with Gaussian noise
subplot(223), imshow(Dout1),title('Mean filtering to Gaussian noise')
% subplot(223), imagesc(Dout1,[0 255]),title('Median filtering to Gaussian noise')
% Mean filtering to image corrupted with Salt and Pepper noise
subplot(224), imagesc(Dout2,[0 255]),title('Median filtering to Salt&pepper noise')

```

Step 10: PSNR

Finally, you may compare the results by using mathematical measure (PSNR – Peak Signal to Noise Ratio). You will compute the PSNR from the images and compare which filtering method is performing better.

PSNR is used to measure the difference between two images. It is defined as:

$$PSNR = 20 \times \log_{10}(b/rms)$$

where “b” is the largest possible value of the signal (typically 255 or 1), and rms is the root mean square difference between two images. The PSNR is given in decibel units (dB), which measure the ratio of the peak signal and the difference between two images.

```

% Compare the gray scale image and image corrupted with Gaussian noise
PSNR(B,C)
% Compare the gray scale image and image corrupted with Salt & Pepper noise
PSNR(B,D)
% Compare the results obtained by Mean filtering on both types of noises
PSNR(I,J)
% Compare the results obtained by Median filtering on both type of noises
PSNR(K,L)

```

Useful Image Processing Commands

Some useful MATLAB commands for image processing. Please type help followed by the function on your MATLAB command prompt to retrieve the information of that function and syntax of use. (e. g. help imread)

- imread
- imshow
- imnoise
- rgb2gray
- filter2
- medfilt2

Sample Results: Sample plotted results are shown in Figs. 5-8.

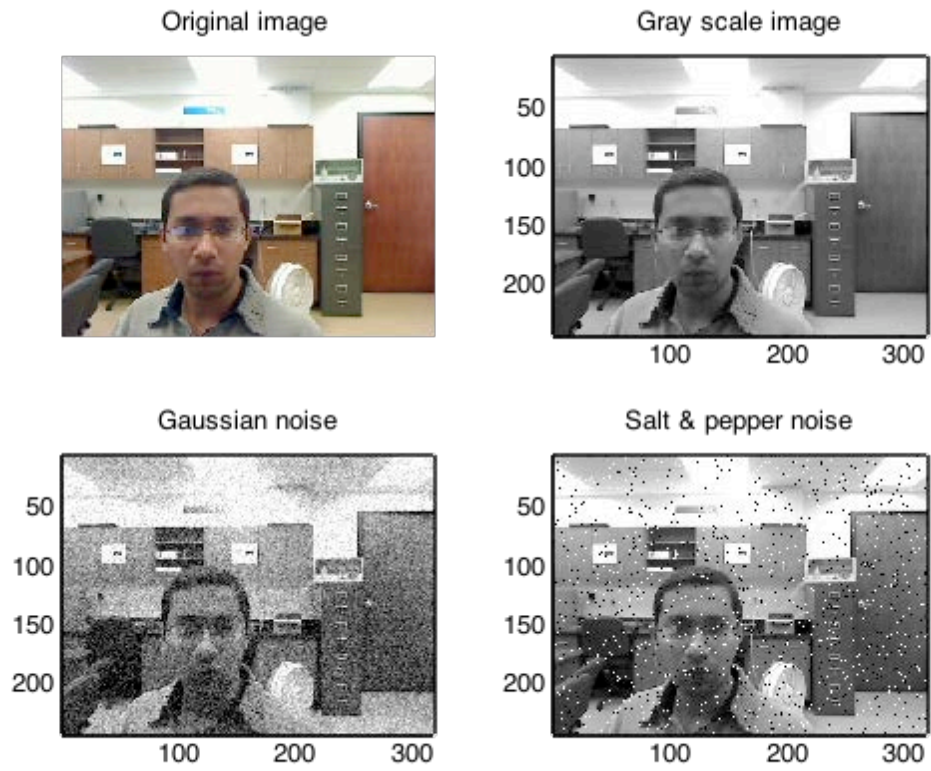


Fig. 5. Image taken by the webcam and two different types of noise added to the gray scale image.



Fig. 6. Image after processing: Mean and Median filtering to remove noise.

PSNR of original gray scale image and image corrupted with Gaussian noise:
PSNR = +28.79 dB

PSNR of original gray scale image and image corrupted with Salt & Pepper noise:
PSNR = +40.13 dB

PSNR after mean filtering:
PSNR = +73.63 dB

PSNR after median filtering:
PSNR = +32.15 dB



Fig. 7. Scanned version of original image (50 ppi), gray scale image of the scanned image, corrupted with Gaussian noise, corrupted with Salt & Pepper noise (clockwise).

Mean filter applied to
Gaussian noise



Mean filter applied to
salt & pepper noise



Median filter applied to
Gaussian noise



Median filter applied to
salt & pepper noise



Fig. 8. Image after processing: Mean and Median filtering to remove noise.

PSNR of original gray scale image and image corrupted with Gaussian noise:

PSNR = +28.76 dB

PSNR of original gray scale image and image corrupted with Salt & Pepper noise:

PSNR = +40.08 dB

PSNR after mean filtering:

PSNR = +73.16 dB

PSNR after median filtering:

PSNR = +35.13 dB

Assessment and Evaluation Plan

Different methods were used for assessment. The participants and their parents, each, were asked to complete separate surveys (The sample surveys are attached in Appendix I). In these surveys, participants were asked about their satisfaction of the camp and if they would be willing to come back next year, and what changes they would like to see in future camps. The participant's evaluation was done through a competition held on the

final day of camp. Prizes were given to the top three individuals. Dr. L. Munski (Executive Director of Dakota Science Center) whose area of expertise is educational foundations of research conducted the surveys. Each year she provided reports to camp organizers to help them improve the camp.

The followings are some of the results she found:

The parent evaluations confirmed that multiple avenues of advertising should be used. These included Grand Forks Herald newspaper, the UND and Dakota Science Center web sites, brochures distributed in local high schools.

Parents were asked what appealed to them the most about the program. All parents responded that the subject matter appealed to them the most. Multiple responses were allowed. The responses are summarized in Table III.

Table III. What appealed to parents most about the program

The major reasons to participate in this camp	Responses
Subject matter	100%
Length of program	27%
Instructors' Credentials	27%
Cost/Value	40%
Daily Schedule	13%
Time of the Day	7%

- When asked if they were satisfied with the helpfulness 80% responded “very satisfied” and 20% responded “satisfied”.
- When asked if they were satisfied with their child’s experience 67% responded “very satisfied” and 33% responded “satisfied”.
- When asked if they would enroll their child in the camp again or an advanced camp 80% responded yes and 6% responded no and 14% did not respond to the question
- Some of the comments that parents provided:
 - My son is so happy about the program. I am already looking forward to another camp like this one. He loved the instructors and is extremely excited to put practice the things he learned at camp. It is really hard to find interesting camps for my son and according to him this is the best he has attended.
 - More introductions to the purpose of MATLAB and examples of real world use of it. Favorite parts of camp were the signal/image activity and circuit lab. Thank you for providing such a unique, hands-on “real engineering” camp.
 - Great camp. My daughter really enjoyed herself. Thanks.

- My son said he did not get some of the content. Maybe the large age that was allowed lent itself to that.
- Thank you for hard work developing this.

Participants were also asked to fill out a survey. The students were asked how satisfied they were with the amount of time spent of each topic. Table IV shows the results.

Table IV. Response appropriateness of time allocated to each topic

	Not Enough Time	Enough Time	Too Much Time	No Comments
Introduction to MATLAB	8.5%	79%	12.5%	0%
Plots in MATLAB	4%	92%	4%	0%
Vectors and matrices	12.5%	83%	4.5%	0%
MATLAB programming	42%	58%	0%	0%
Sound processing	16%	80%	0%	4%
Image processing	17%	83%	0%	0%
GUI	20%	80%	0%	0%
Problem of the day	12.5%	83%	0%	4.5%

When students were asked about the final competition, 92% found it “Challenging” and 8% found it “Too Difficult”. None of them found it “Not Difficult”. Also 92% found the booklets helpful.

Almost all students wanted either to attend a similar camp and/or refer their friends to this camp.

The student evaluations were taken with anonymity therefore a correlation cannot be made between the responses and the demographic information. The demographic information of the camps is available. The distribution in terms of the grade levels is depicted in Fig. 9.

In summary, the camp was well received and students showed enthusiasm in continuing to use the material in future. However, no tracking was performed if they chose Engineering as their field in college and how the camp affected their choice.

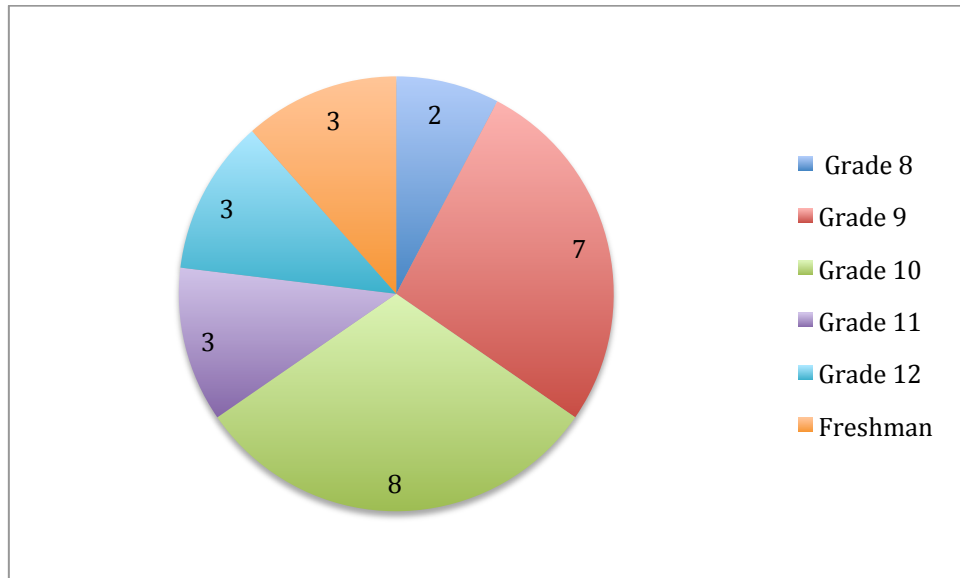


Fig. 9. Grade level distribution of camp participants for 3 years (2010-2012)

Conclusions and Lesson Learned

In general, those who participated in the camp were satisfied and felt that it was fun and informative. The option of residential camp was offered in 2011 and 2012, for those who were coming from out of town. In future, we should consider some interesting activities to fill out the evening time. The option of combining the camp with some field trips to local industries in engineering fields is also considered for future.

The instructors were happy to see that young attendees were able to grasp the concept quickly. They become so comfortable with the programming that some of them during the challenges even started adding extra parts to their program and create movies or different options that were not asked as a part of their evaluations.

Acknowledgment

Financial support from the University of North Dakota Summer Program and Events Council is gratefully acknowledged. The College of Engineering and Mines provided technical and computer supports. The IEEE Red River Valley Section and MathWorks provided prizes for the competition winners.

Bibliography

1. MathWorks, <http://www.mathworks.com>
2. G.C. Weaver, K. Haghghi, D.D. Cook, C.J. Foster, S. M. Moon, P.J. Phegley, R. L. Tormoehlen "Attracting Students to STEM Careers, A white paper submitted to the 2007 - 2013 Purdue University strategic planning steering committee," [online], available at http://www.purdue.edu/strategic_plan/whitepapers/STEM.pdf.

3. G. Rosen, J. Silverman, and A. Chauhan, "Connecting artistically-inclined K-12 students to physics and math through image processing examples," in *Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop, 2009. DSP/SPE 2009. IEEE 13th*, 2009, pp. 419-424.
4. G. Rosen, M. Usselman, and D. Llewellyn, "Relating high school mathematics through sound and images," in *ASEE Southeast Section Conference*, 2005, pp. 1-10.
5. S.C. Douglas, "The INFINITY project: Digital signal processing and digital music in high school engineering education," *2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*, New Platz, NY, October 2001.
6. J.H. McClellan, R.W. Schafer¹, M.A. Yoder, "Experiences in Teaching DSP First in the ECE Curriculum," *ASEE Annual Conference Proceedings*, Milwaukee, WI, June 1997.

Appendix I: Evaluation Form Samples



School of Engineering and Mines
The University of North Dakota
Grand Forks, North Dakota
701-777-3390

Youth Participant Evaluation Form

Program Title: MATLAB Mathematics & Engineering
Instructor s: Dr. Reza Fazel & Dr. Sima Noghianian
Dates: July 30 – August 2, 2011 *Sponsored by the UND Summer Programs and Events Council*

1. What was your favorite part of the program? _____
2. What was your least favorite part of the program? _____
3. Overall were the daily instruction booklets helpful? YES NO
4. Did your teachers present the information in a way that you could understand it? YES NO
5. Would you like to attend similar UND Summer Programs? YES NO
6. Would you tell your friends about UND MATlab Mathematics & Engineering? YES NO
7. How satisfied were you with the amount of time spent on the following topics (circle one):

Introduction to MATLAB	Not Enough Time	Enough Time	Too Much Time
Plots in MATLAB	Not Enough Time	Enough Time	Too Much Time
Manipulation of vectors & matrices	Not Enough Time	Enough Time	Too Much Time
How to write MATLAB program	Not Enough Time	Enough Time	Too Much Time
Sound Processing in MATLAB	Not Enough Time	Enough Time	Too Much Time
Image Processing in MATLAB	Not Enough Time	Enough Time	Too Much Time
Demo of the Day Which demo was your favorite?	Not Enough Time	Enough Time	Too Much Time
Nerosky Mind Games Mindflex	Electronic Kits	Voice and Light Circuit Lab	
Challenge of the Day	Not Enough Time	Enough Time	Too Much Time
Final Competition	Not challenging	Challenging	Too Difficult
How satisfied were you with the food	Fair	Good	Excellent
How satisfied were you with the residence hall (if you used it)	Fair	Good	Excellent
8. Are there any comments that you would like to share regarding the classroom and computer lab? _____
9. Is there anything else you would like to tell us? _____

Student Name: _____

1. How did you hear about this program? School Parents Friends Other _____

2. What interested you most about this program? _____

3. What grade will you be entering this fall? _____
4. Name of school _____ Home Schooled
5. What math class did you complete last year in school? _____
6. Have you taken computer classes in school? YES NO
7. If you answered yes to question 5 please provide a short description of the computer classes. _____

8. What interests would you like to pursue in college? _____

9. When thinking about a career would you like to work on a team full time or part time? (circle one)
10. Service Learning is a volunteer opportunity where you can experience a career field such as engineering and complete a community service project.
Would you like to participate in Service Learning? YES NO

Sponsored by the UND Summer Programs and Events Council