AC 2011-2398: USING DIGITAL IMAGES TO TEACH ABSTRACT MATH AND INSPIRE STUDENTS TOWARDS CAREERS IN COMPUTER SCIENCE AND ENGINEERING

Victor Mejia, California State University, Los Angeles  
Jessica Alvarenga, California State University, Los Angeles  
Jianyu Dong, California State University, Los Angeles  
Huiping Guo, California State University, Los Angeles  
Israel Hernandez, California State University Los Angeles

Mathematics teacher for STEM at Roosevelt High School. MESA advisor.

Eun-Young Kang  
Mr. Phanit Pollavith  
Adriana Trejo, Roosevelt High School  
Nancy Warner-Perez, California State University, Los Angeles

Nancy Warner-Perez is a professor of Electrical and Computer Engineering at California State University, Los Angeles and the director of the IMPACT LA NSF GK12 Program.
Using Digital Images to Teach Abstract Math and Inspire Students towards Careers in Computer Science and Engineering

1. Introduction

Abstract mathematical concepts are often difficult for high school students to understand. Students are often adept at learning how to solve specific math problems but have a difficult time understanding how they can relate to the real world. In response to this, math teachers try to come up with engaging, real-world examples to not only allow students to better understand abstract mathematical concepts but also to reinforce their comprehension. These real-world examples must meet state standards and should be designed to increase students’ interest in science and engineering.

Image processing is a real-world example of engineering that students easily relate to because of their familiarity with cell phones, video game consoles, cameras, and computers. This paper presents a collection of different demonstrations and hands-on activities that were developed to help students learn about digital images while reinforcing their understanding of abstract mathematical concepts such as matrices and functions. Through the IMPACT LA GK-12 Program, Improving Minority Partnerships and Access through CISE (Computer & Information Science & Engineering)-related Teaching, three graduate fellows were paired with three different high school math teachers and were successfully able to engage students in their graduate research through these demonstrations and hands-on activities. Through the IMPACT LA Program, graduate student fellows become educators, thus bridging the gap between abstract math and real-world engineering by bringing graduate student research into the classroom.

The three different areas of research that were brought into the classrooms are: computer vision and object tracking, information security and digital watermarking, and telescopes and deep-space exploration. How can a computer track moving objects in a video? How can you detect if a video has been tampered with? How are pictures sent on your cell phone or from the Hubble Space Telescope? These are some of the research-related questions that students were able to explore through demonstrations and hands-on activities.

This paper is organized as follows. Section 2 describes the IMPACT LA Program and its goals. Section 3 introduces the fellows’ research in computer vision and object tracking, information security and digital watermarking, and space telescopes and deep-space exploration. Section 4 describes the mathematical concepts the high school math students were learning when these demonstrations and activities were done and presents the mathematical background needed to understand them. Section 5 presents the demonstrations that were used. It first describes the multimedia that was used to introduce students to space and satellites, and then presents the science and engineering tools used to introduce students to digital images and object trajectories. Section 6 describes two different hands-on activities used, digital image decoding and assembly of macroblocks, along with an explanation of how the activities reinforce concepts of matrices and algebraic functions. Students’ evaluations of the hands-on activities are also presented that demonstrate students’ understanding of abstract math concepts. Finally, section 7 presents pre- and post-assessment data that demonstrate noteworthy improvements in attitudes of students towards computer science and engineering, respectively.
2. IMPACT LA GK12 Program Information

The IMPACT LA Program partners graduate teaching fellows with middle and high school math and science teachers in the Los Angeles Unified School District (LAUSD). The program is centered at California State University, Los Angeles (CSULA), and is part of the National Science Foundation (NSF) Graduate STEM Fellows in K-12 Education (GK-12) Program, which provides fellowships and training for graduate students in science, technology, engineering, and mathematics (STEM). The graduate student fellows serve as visiting scientists or engineers who work closely with their partner teachers to engage middle and high school students in science and engineering demonstrations, presentations, and activities related to the fellows’ research.

The two primary goals of the IMPACT LA Program are to 1) change teachers, students, and parents’ perceptions of engineers and encourage K-12 students to explore engineering and research careers, and 2) to enhance the communication and research skills of graduate fellows. To achieve these goals, during workshops teachers participate in a wide range of research experiences designed by fellows to introduce and update teachers to their research areas. Graduate fellows conduct Master’s thesis research in Computer Science, Computer/Electrical Engineering, Bioinformatics, Biomedical Engineering, Mechanical Engineering, Civil Engineering, Biology, Chemistry, Physics, and Math. CSULA faculty train fellows through a preparation course and workshops in order to improve communication, collaboration, and teaching skills. Furthermore, a strong partnership between CSULA, LAUSD, local industry, and minority serving organizations such as Great Minds in STEM and MESA has been established in order to achieve program goals. Broader impacts include increasing the number of underrepresented minority students who pursue college degrees and careers in STEM and to strengthen the research and teaching skills of the graduate fellows.

At the time these demonstrations and activities were performed, the program consisted of eight fellows, conducting research in Electrical Engineering, Civil Engineering, Chemistry, Computer Science, Biology, and Biochemistry. A total of nine teachers are in the program and all but one fellow are partnered with an individual teacher. In the IMPACT LA Program, fellows are required to spend 10 hours per week in the classroom conducting presentations, activities, and demonstrations related to their research and science and engineering. Fellows are partnered with teachers throughout the entire school year. Currently, three fellows are partnered with three high school math teachers while the rest are placed in middle school math and science classes. Two of the fellows in high school math classes are Computer Science graduate students, while the other is an Electrical Engineering graduate student. This paper focuses on the research of these three fellows and how it was infused into Algebra II and Pre-Calculus curriculum through hands on activities and demonstrations.

In addition to providing in-class hands-on activities, fellows along with their partner teachers are required to write and publish four lesson plans, conforming to the TeachEngineering.org guidelines. Care is taken when writing lesson plans to include sufficient information for teachers including connections to engineering/science, background information for teachers about both the subject matter and the engineering/science research-related concepts, cost, materials, etc. This is done in an effort to maintain the activities so that any teacher can re-
use them and adapt them to their specific needs. As a result, when the fellow is no longer available, their partner math teachers and other teachers can take advantage of these valuable lesson plans. Lesson plans, worksheets, and demonstrations are posted on the program website (impactla.calstatela.edu) and selected lessons will be submitted to TeachEngineering.org.

3. Fellows’ Research Overview

a. Computer Vision and Object Tracking

Humans think in terms of objects. Objects surround us, and we describe the world in terms of objects. Computer Vision applications are found in everyday items such as the face detection in today’s digital cameras and the optical character recognition in the envelope-free deposit ATMs. Computer Vision is the study that allows computers to understand images just as humans do.

Object detection and tracking in both images and video is an important task in Computer Vision applications. Robotic navigation and obstacle avoidance, video surveillance, and traffic monitoring are noteworthy applications of Computer Vision. In simple terms, object tracking is defined as the problem of estimating the trajectory of an object in a video sequence. Most of the current object tracking algorithms have specific applications such as face-detection, tracking cars, people, etc.

This research focuses on developing an algorithm for tracking multiple moving objects with occlusions in high-resolution video sequences. Rather than focus on a specific type of object, this research covers general moving objects. Furthermore, most of current algorithms also assume a stationary camera in which the background can be learned over a long period of time and usually consists of tracking objects in a very low-resolution video sequence. This research does not require learning the background and focuses in tracking multiple moving objects in high-resolution video. The proposed object tracking will be integrated during the decoding stage of the H.264 compression, the current state-of-the-art compression standard, in order to take advantage of its robust motion estimation, a really useful feature which provides us with an initial estimate of the detected moving objects.

b. Digital Watermarking

Today, digital data such as images, videos, and audio are easily created, modified, reproduced, and redistributed. It is difficult to distinguish between authentic digital data, and digital data that has been modified or tampered with. It is essential to protect the authenticity of digital data. Digital Watermarking is the process of embedding information into a signal in a way that is difficult to remove. The process of embedding a digital watermark can be applied to digital data such as images, videos, and audio. When the digital data with an embedded watermark is copied, the embedded watermark is copied along with it.

There are two types of digital watermarks, visible watermarks, and invisible watermarks. A visible watermark is a transparent image or text that is overlaid on the primary image. An invisible watermark is an algorithm that is applied to an image; an invisible watermark cannot be
seen by the human eye but can be detected algorithmically. This research on digital watermarking focuses on invisible watermarking on digital images and videos.

There are two main components in developing invisible digital watermarks for digital images and videos. The first component consists of embedding the watermark. A watermark embedding algorithm is executed on the original digital data to produce an output image containing the watermarked digital data. This watermarked data will not have any visible degradation and will be identical to the original according to the human eye. The second component consists of detection of the watermark. In order to ensure authenticity, the watermarked data is passed through a watermarked detection scheme. If the watermark still exists, and has not been damaged, we are ensured that the digital data has not been tampered with. The purpose of digital watermarking is to ensure authenticity of the digital data. This includes tamper detection, integrity proof, and detecting modifications to digital data.

c. Space Telescope and Deep-Space Exploration

With the increasing need to see further into space and get closer to answering questions about our universe, it is necessary to create a space telescope capable of collecting light closer to when the Big Bang occurred. Currently, NASA is developing the James Webb Space Telescope (JWST). The JWST, the successor of the Hubble Space Telescope (HST), is scheduled to launch in 2014. The purpose of space telescopes such as the HST and the JWST is to capture the light from the beginnings of our universe. The focus of this research is in the control of large scale space structures, specifically a segmented telescope testbed model of the JWST.

In order to construct a telescope to the scale required to see further back into the universe, it is necessary to increase the size of the primary mirror. However, it is both extremely difficult and costly to make a single mirror to the accuracy and precision necessary to collect images. Unlike the HST, which employs a single monolithic parabolic mirrored dish to collect light, the JWST would require a mirror that would not be transportable using the current spacecraft. In addition, the JWST would need to be placed at an L2, Lagrange orbit so that it may collect light from objects whose light has shifted into the infrared spectrum. A solution is to segment the mirror into smaller mirror segments to approximate a parabolic shape that may then be actively controlled to maintain its parabolic shape and allow for compaction for transport. Control algorithms can also be developed to adjust for noise, disturbances and fault detection.

In the NASA University Research Center (URC) Structures, Propulsion and Control Engineering (SPACE) Laboratory, research is being conducted on a testbed version of the JWST. This fellow’s research explores using control algorithms, such as an H-infinity controller, to perform control of the mirror shape, adjust for disturbances, and allow for precision pointing of the telescope at a simulated source target.

4. Background Mathematical Concepts

The hands-on activities and demonstrations presented in this paper reinforce some abstract mathematical concepts covered in Algebra II and Pre-Calculus including matrices,
function transformations, and the quadratic equation for an object in free fall. A quick review of
these concepts is provided to demonstrate the material students were learning during the time the
demonstrations and hands-on activities were performed.

a. Matrices

A matrix is a rectangular array of numbers represented by one or more rows, and one or
more columns. Although a matrix may contain many numbers, each matrix is treated as a whole,
and can be manipulated according to its rules. Some of the basic operations include addition,
subtraction, and scalar multiplication. In matrix addition, and subtraction, two matrices are added
or subtracted. This is done by adding or subtracting the corresponding entries together. In
addition and subtraction, the matrices should have the same dimensions in order to perform the
operation.

b. Functions

Function Transformations

In mathematics, there are several basic functions: \( f(x) = c \), where \( c \) is a constant, \( f(x) = x \),
\( f(x) = x^2 \), \( f(x) = x^3 \), \( f(x) = |x| \), \( f(x) = \frac{1}{x} \). Various transformations or combination of
transformations can be performed on a basic function. Transformations can cause a shift, a
reflection, a stretch, or shrink of the original graph. For example, we can negate a function such
that \( g(x) = -f(x) \), which will produce a reflection across the x-axis. We can have \( g(x) = f(-x) \),
which will produce a reflection across the y-axis. We can modify the function such that \( g(x) = f(x) + c \), which will shift the original graph up by \( c \) units, or \( g(x) = f(x-c) \), which will shift the
original graph of \( f(x) \) to the right by \( c \) units. These transformations do not change the shape of
the graph. However, there are also transformations that changes the shape of the graph, such as
\( g(x) = 2f(x) \), or \( g(x) = f(3x) \).

Quadratic Functions

Another mathematical concept used in the demonstrations is quadratic functions.
Students first learn that the height of an object in free fall follows this quadratic function:

\[
h(t) = -\frac{1}{2}gt^2 + v_0t + h_0
\]

Where, \( h(t) \) is the height of the object (in meters) at any given time \( t \) (seconds), \( v_0 \) is the initial
velocity (meters per second), \( h_0 \) is the initial height, \( t \) is time, and \( g \) is the gravity constant,
which is 9.8 m/s\(^2\).

The time it takes the object reach its maximum height is given by

\[
t = \frac{v_0}{g}
\]

Given any initial height and initial velocity when the ball was thrown, students can determine the
height at any given time \( t \) using this quadratic equation.
5. Demonstrations

a. GPS and Google Earth: Learning about how digital images are used by satellites

While space telescopes allow us to see objects in the distant universe, there also exist satellites that can take images of our earth. These images can be used to create composite images of earth, or to map small areas. These images can then be used in such applications as Global Positioning System (GPS) navigation, maps of the terrain, and even to reconstruct large areas of the earth. In order to show students how individual, as well as scientists and engineers, can take advantage of this technology, students were introduced to GPS and Google Earth. These demonstrations allowed students to see the types of images that can be captured by space satellites. In addition, Google Earth allows students to see the earth in ways that they may not have previously been able to living in a major city.

To give the students something that would interest them, a bird’s eye view image of their high school was shown to them using Google Earth. They were able to identify their classroom as well as other landmarks on the map. They were then introduced to latitude and longitude and how GPS is able to pinpoint locations on earth given their coordinates.

Students were then asked to think about how satellites transmit data down to earth. The concepts can be explained to the students in the following manner. In the beginning of the life of the camera, film was and is still being used to capture light and create a picture. Polaroid is another type of technology that allowed for immediate development of images rather than having to take your film to be developed at the 1-hour photo mart. Today, technology has allowed for the creation of digital cameras. These cameras convert light captured by the camera’s lenses into a number that can be stored digitally on the camera or small storage device. We can then transfer this data onto our computers and view the results on our monitors. In order to properly display the image, there must be a system for the computer to translate this data into some quantifiable intensity of light and color. The way this is accomplished is to use RGB values. RGB stands for red, green and blue. This data is then transmitted using radio frequencies down to earth where it is decoded by computers and interpreted into the images we see on our screens. In the next demonstration, students learn how matrices of RGB values are used to represent images.

b. Science and Engineering Tools

MATLAB and Java

Fellows, as visiting scientists and engineers in the high school classrooms, use state-of-the-art engineering tools in their demonstrations such as MATLAB (MATrixLABoratory), a numerical computing environment, and Java (a computer programming language). To introduce digital image representation, students were first asked to think about how computers, which include their own personal cell phones, store pictures they take. To increase student participation, an image of the students, shown in Figure 1, was taken using a consumer digital camera.
The image was then loaded to the MATLAB environment and shown to the students (refer to Figure 2). In MATLAB, the students could see that the size of the image was given as 480x640x3. Students understood where the 480x640 came from (the image’s height and width), but not the additional 3. After a little speculation, students concluded that it had to do with color. The simple concept of primary colors (red, green, and blue) applies to digital image representation, since computers store images as three different matrices, where each element is a pixel and each two-dimensional matrix represents a color image (red, green, and blue). Using simple MATLAB commands, individual red, green, and blue matrices were extracted and the corresponding images shown to the students.

Figure 1. Digital image used in demonstration.

Figure 2. MATLAB environment and a computer’s digital image representation as three individual color matrices.
A second demonstration was used to explain to students how computers can change the color scheme of an image to grayscale, sepia, etc. A grayscale representation of an image is simply a linear combination of the three color matrices:

$$\text{Gray} = 0.29 \ast \text{Red} + 0.59 \ast \text{Green} + 0.11 \ast \text{Blue}$$

The grayscale now has size of 480x640, and it is only one matrix. Using the programming language Java (though it can also be done in MATLAB), an image was loaded into memory and the grayscale image was computed and then saved back into the computer as shown in Figure 3.

Figure 3. Java programming and MATLAB were used to show students how to manipulate images such as change to grayscale.

Demonstrations using Java and MATLAB showed students how math functions can be used to transform color images into grayscale images. Furthermore, students were shown how to manipulate matrices using MATLAB to modify the image color, and the image intensity. As students study the mathematical concept of composition of functions, they were shown a real world example of composition of functions using MATLAB. Students were first shown an original image. Then together as a class, we wrote a function that takes in one parameter (an image), and outputs a grayscale image. After that, they wrote another function that also takes one parameter (an image), and multiplies all of the values inside the matrices in the parameter by 2. By doing this, the values of the matrices of this parameters will double. If the parameter passed in is a grayscale image, then this will result in increased brightness.

After we wrote the two different functions, we called the function that outputs a grayscale image $f$, and we called the function that multiplies the values of the matrices by 2, $g$. We call the original image $x$. We executed $f(x)$ by passing the original image through the function $f$ that we wrote. The result of $f(x)$ is a grayscale image. We then executed $g(f(x))$, which increases the intensity of $f(x)$, which is the parameter passed into the function $g$. As a result, the original image
that we passed in was first converted to grayscale, and the brightness level increased. Students are able to gain a better understanding of functions and how they are applied to real world problems. Students are also exposed to computer science concepts of writing functions, passing in parameters, and Java and MATLAB programming.

**Python program and Maya**

Two other science and engineering tools used were Maya (a 3D computer graphics software) and Python (another programming language). Once students learned the digital representation of images, they learned that a video is composed of numerous frames, each frame being an individual image, and each video usually composed of 30 frames per second. In order to visualize an object trajectory in a video, the class was led through a demonstration of how to use the Python programming language, Maya, and the quadratic formula to create an animation of a bouncing ball. Through this, the students could visualize how the quadratic formula computes the height of an object in free fall and how programming languages can be used to create applications such as those used in gaming systems.

To create the animation, the position of the ball should automatically be recorded for each frame. The height formula computes the height at any given point in time given initial velocity and gravity. The value of $t$ will vary from 1 up to a certain number of frames. To do so, the scripting language Python is used, and a frame is recorded at every value of $t$.

For demonstrations purposes in the Maya 3D graphics environment, an initial height of 2 meters, initial velocity of 1.6 meters, and a gravity constant of -0.04 meters/s$^2$ was used (as shown in Figure 4). From this we can calculate the amount of frames it takes to reach the maximum height, which is $t = \frac{v_0}{g} = \frac{1.6}{0.04} = 40$ frames. This means that it will take another 40 frames for the object to come back down, for a total of 80 frames. The height formula $h(t) = \frac{1}{2}gt^2 + v_0t + h_0$ was incorporated into the Python code:

```python
import maya.cmds as cmds
cmds.polySphere( name = 'bouncing', radius = 2 )
g = -0.04
v0 = 1.6
y0 = 2
for ty in range(1,80):
    h = 2 + 1.6*ty - 0.04*ty*ty/2
    cmds.setKeyframe('bouncing', attribute='translateY', value=h, t=ty)
```

The first two lines of the code simply allow Maya to create a spherical object named “bouncing” with a radius of 2 meters. Next we define the constants in the next three lines. The snippet of code “for ty in range(1,80):” means that the variable ty will take on the values of 1,2,3, etc., all the way to 80. For each value of ty, the height will be calculated using the height formula, and it will record a frame for the given height. After these calculations, the animation can be played continuously to show a bouncing ball.
6. Digital Image Based Hands-on Activities

a. Digital Image Decoding: Assembling images based on color data in a matrix

In order to display an image on a computer monitor or newer television screens, the image must be divided into various pixels. The higher the number of pixels in an image, the better the resolution, and in turn, the higher the quality of the image. Resolution is represented as the number of rows of pixels by the number of columns of pixels in which the image has been divided. Each pixel in a digital image consists of a single color that can be represented in terms of some intensity of red light, green light and blue light. We are used to red, blue and yellow as primary colors. However, colors of light mix much differently than mixing colors of paint. In light, red, green and blue are primary colors and can be used to create the entirety of the visible spectrum as shown in Figure 5. The color of light is dependent on its wavelength. White light consists of all the colors combined as shown in Figure 6.
We have discussed how images are represented by three matrices, for red, green, and blue light intensities. But, how are each of these intensities represented in a digital image? Digital devices store data in a binary format. Binary is base-2 representation which uses only two digits, 1 and 0. In binary, the location of the digit represents a power of 2 while the value of the digit (1 or 0) represents a constant of multiplication. For example, the binary value:

10110001

represents the decimal value 177. This can be calculated by adding all powers of 2 in which a 1 is present. The far right digit represents \(2^0\). The next digit over represents \(2^1\). This continues until you reach the final digit, in this case \(2^7\),

$$
\begin{array}{cccccccccc}
\text{Binary Digit} & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\
\text{Power of Two} & 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\
\end{array}
$$

then,

$$
10110001 \text{ (binary)} = 2^7 + 2^5 + 2^4 + 2^0 = 128 + 32 + 16 + 1 = 177 \text{ (decimal)}.
$$

Image color intensities can be represented by an 8-bit integer. This will result in 256 distinct values from 0 to 255. A computer represents the intensity of red, green and blue light in each pixel by this 8-bit integer. A value of 0 means no light and a value of 255 means the brightest light. Since white light consists of all the colors, it is represented by the (R,G,B) value (255, 255, 255). A black pixel would then be the absence of light and would be represented by (0, 0, 0). Shades of gray may be created by using equal values of red, green and blue, and other colors are created by using varying values of each.
For this activity, students reconstructed an image based on the RGB values given to them in various matrices. They were given the red, green and blue matrices of a portion of an image as well as a RGB color code table. In order to determine the color of each pixel, students had to look at the RGB values of each pixel and find the appropriate color in given table. They then had to label the color of each pixel into their Image Grid.

<table>
<thead>
<tr>
<th>Color</th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yellow</td>
<td>255</td>
<td>235</td>
<td>49</td>
</tr>
<tr>
<td>Red</td>
<td>225</td>
<td>55</td>
<td>30</td>
</tr>
<tr>
<td>Blue</td>
<td>83</td>
<td>102</td>
<td>221</td>
</tr>
<tr>
<td>Brown</td>
<td>151</td>
<td>85</td>
<td>45</td>
</tr>
<tr>
<td>White</td>
<td>255</td>
<td>255</td>
<td>255</td>
</tr>
<tr>
<td>Black</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Beige</td>
<td>244</td>
<td>202</td>
<td>158</td>
</tr>
<tr>
<td>Light Blue</td>
<td>162</td>
<td>246</td>
<td>255</td>
</tr>
<tr>
<td>Green</td>
<td>32</td>
<td>191</td>
<td>0</td>
</tr>
<tr>
<td>Orange</td>
<td>223</td>
<td>154</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 7. Lookup table for RGB values.

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
<th>Image Grid</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>235</td>
<td>49</td>
<td>[Yellow  Black]</td>
</tr>
<tr>
<td>32</td>
<td>191</td>
<td>49</td>
<td>[Green  Yellow]</td>
</tr>
</tbody>
</table>

Figure 8. Decoding of pixels.

After the students finished decoding their image, they were then able to assemble their image by placing the appropriately color pixel into the appropriate place in the image. Pixels were simply squares of colored construction paper or card stock. Each group was given a portion of the image to construct. As each group finished their portions, they placed their piece onto a poster displayed to the class as shown in Figures 9 and 10.

Figure 9. Image as each group completes their piece.
Assessment Data of Image Decoding Hands-on Activity

After students engaged in the hands-on activity regarding image decoding, survey responses were collected. Figure 11 shows the impact of this particular hands-on activity. Overall, approximately 77% of the students who engaged in the activity did not know how computers store and decode images prior to the activity. A total of 97% of students reported that the activity helped them improve their math skills with matrices, and 100% reported that the activity helped them understand how matrices are used in real life.

Figure 11. Students’ attitudes towards the image decoding activity (n = 35, students).

Figure 12 shows the students’ difficulty rating. Based on an increasing scale of “fun” and “hard” from 1 to 5, students gave an average fun rating of 4.29 and an average difficulty rating of 1.89. These figures show that this engaging activity was both fun and effective in teaching
students about real-world engineering and helping them boost their understanding of abstract mathematical concepts.

![Bar chart showing student difficulty rating](image)

Figure 12. Students’ difficulty rating of the image decoding activity (On a level from 1 to 5, how fun was this activity? On a level from 1 to 5, how hard was this activity?) (n=35 students).

b. **Macroblocks in Digital Images: Applying math functions to identify and assemble them**

Students further enhanced their understanding of the computer science concept of digital image representation through a hands-on activity that introduced the concept of macroblocks and reinforced their understanding of function transformations. Pixels are the smallest units in an image, and adjacent pixels are grouped together to form macroblocks. Techniques such as digital image watermarking are applied to macroblocks of an image so that if part of the image is lost or tampered with, the watermark still remains in other macroblocks of the image. In Figure 13, each group constructed a portion of the overall image. There were a total of six separate sections being constructed. Each of those sections can be considered a macroblock.

In this activity, students applied the mathematical concept of function transformations in order to put together an image from a worksheet of scrambled macroblocks. First, students reviewed the basic algebraic functions and their graphs. Then, students were taught about rigid and non-rigid transformations that can be made to transform those algebraic functions. Two separate worksheets were constructed for this activity. The first worksheet contains an algebraic function on the top left, and nine different mathematical transformations or combination of transformations. Students then rewrote these transformations in function notation and wrote the corresponding answers directly into the box.
After each group has been given sufficient time to rewrite the transformations in all the boxes, they were given the second handout shown in Figure 14. This handout contains answers to the worksheet in scrambled order. Under each answer was a macroblock corresponding to part of an image. Students then matched the function notation they had solved in Figure 13. Students then cut out the macroblock corresponding to the answer, and glued the macroblock over the worksheet.
Each group of students repeated this process until all worksheets were completed, and a larger macroblock was formed. At the end of the class, all of the macroblocks that each group produced were put together to form one complete image as shown in Figure 16. This activity exposed students to concepts of pixels and macroblocks while learning about function transformations.

Figure 16. Image of completed macroblocks put together to form complete image.

Figure 17. Image of students working on assembling the scrambled macroblocks.
Assessment Data of Macroblocks Hands-on Activity

Figure 18 shows the impact of this particular hands-on activity. Approximately 60.98% of the students who engaged in this particular activity did not know how computers store and decode images prior to the activity. A total of 90.24% of students indicated that the activity helped them improve their math skills with function transformations, and 87.8% reported that the activity helped them understand how macroblocks are used in real life.

![Bar chart showing students' attitudes towards the macroblocks activity (n=41).](chart1)

Figure 18. Students’ attitudes towards the macroblocks activity (n=41).

Figure 19 shows the students’ difficulty rating. Based on a scale from 1 to 5, students gave a fun rating of 3.85 and a difficulty rating of 2.7. These figures show that although the activity may have been a somewhat difficult for some students, it was fun and helped students understand how computer process images in terms of macroblocks while at the same time they increased their understanding of abstract mathematical concepts.

![Bar chart showing students' difficulty rating of the macroblocks activity (n=41).](chart2)

Figure 19. Students’ difficulty rating of the macroblocks activity (n=41).
7. Conclusion and Pre- and Post-Assessment Data

The digital image demonstrations and activities presented in this paper reinforced the concepts being covered on matrices and functions in Algebra II and Pre-Calculus. The demonstrations of satellites and GPS and the software tool MATLAB allowed students to obtain an initial glimpse of the computer representation of images and how computers use matrices to store them. The hands-on activities allowed students to see how the operations that they apply to matrices are also computed by computers. The activity with macroblocks allowed students to further their skills with function transformation and understand how pixels can be grouped into macroblocks for image processing techniques such as digital watermarking.

Perceptions of Engineering

The digital image demonstrations (with the exception of the GPS/Google Earth demo) and the image decoding hands-on activity have been conducted in Algebra II and Pre-Calculus classes in Roosevelt High School for the past two years. Pre and Post-surveys are collected annually in the IMPACT LA program to gauge students’ attitudes towards math, engineering and science, since it is a primary goal of the program to increase student interest. The data was collected for all students participating in the program including middle and high school math and science classes. Figure 20 shows the impact of the IMPACT LA Program on the students’ attitudes towards engineering and science for the 2009-10 school year. Through their interactions with the students, fellows had a relatively significant impact on students’ view on engineering. We can see that the activities and demonstrations conducted in math classes had a more significant impact than in other subjects. More students thought that engineering was fun and they enjoyed studying it more.

![Figure 20. Differences in attitudes across classroom subjects.](image)
In addition, the survey also asked students to list three words that describe a scientist or engineer. Table 1 shows the answers, with the top five being smart, hard-working, intelligent, creative, and invent. It was encouraging to see the increase in positive attributes of scientists and engineers (“creative” and “invent”) and a decline in the negative attributes (“nerd” and “boring”).

<table>
<thead>
<tr>
<th>Responses</th>
<th>Pre-Assessment</th>
<th>Post-Assessment</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart</td>
<td>44%</td>
<td>50%</td>
<td>+6%</td>
</tr>
<tr>
<td>Hard-working</td>
<td>12%</td>
<td>14%</td>
<td>+2%</td>
</tr>
<tr>
<td>Intelligent</td>
<td>10%</td>
<td>12%</td>
<td>+2%</td>
</tr>
<tr>
<td>Creative</td>
<td>5%</td>
<td>9%</td>
<td>+4%</td>
</tr>
<tr>
<td>Invent</td>
<td>2%</td>
<td>5%</td>
<td>+3%</td>
</tr>
<tr>
<td>Cool</td>
<td>7%</td>
<td>11%</td>
<td>+4%</td>
</tr>
<tr>
<td>Awesome</td>
<td>1%</td>
<td>5%</td>
<td>+4%</td>
</tr>
<tr>
<td>Nerd</td>
<td>3%</td>
<td>1%</td>
<td>-2%</td>
</tr>
<tr>
<td>Boring</td>
<td>3%</td>
<td>0%</td>
<td>-3%</td>
</tr>
<tr>
<td>I don’t know</td>
<td>8%</td>
<td>1%</td>
<td>-7%</td>
</tr>
</tbody>
</table>

Table 3: Descriptions of Scientists and Engineers.

Conclusions and Future Work

Since 2008, the IMPACT LA Program has been placing graduate student fellows in middle and high school classrooms in order to increase students’ understanding of science and engineering. This has been achieved through the fellows’ introductions of demonstrations and hands-on activities that not only increase student understanding of abstract mathematical concepts but also positively change their perception of science and engineering. In order to preserve the activities and demonstrations when the fellow is no longer working with their partner teacher, lesson plans, including worksheets and demonstrations, are available on the program website. An emphasis has been placed on lesson plan writing this year to ensure that the lessons are clear and can be widely adopted by teachers at other schools. In the future, we also plan to evaluate the partner teachers’ abilities to continue implementing these lesson plans once their yearly commitment ends. In order to broaden the impact of these activities, fellows are also encouraged to volunteer at other local schools through the National Lab Day program in order to perform and distribute the activities and lesson plans.

Acknowledgement

This work is supported by NSF Grant #0742426.
Bibliography


