Connecting Forensics and Linear Algebra

http://ichef.bbci.co.uk/images/ic/480xn/p025wxrb.jpg

By
Donna Beers, Simmons College, donna.beers@simmons.edu
Catherine Crawford, Elmhurst College, crawford@elmhurst.edu
Connecting Forensics and Linear Algebra

Summary of the Module
This module aims to enrich a first course in undergraduate linear algebra by introducing students to the concepts of linear combination, spanning set, linearly independent set, and basis through engaging real-world examples drawn from the field of forensics. The module also explores the variety of career options in forensic science.

Note to teachers: Teacher notes appear in dark red in the module, allowing faculty to pull these notes off the teacher version to create a student version of the module.

Target Audience: Sophomore and junior mathematics majors as well as students majoring in the natural sciences, economics, and computer science.

Prerequisite: Students should be currently be enrolled in a linear algebra course and be familiar with the notion of a vector space and with standard examples of vector spaces including:

- \( n\)-dimensional Euclidean space, \( \mathbb{R}^n = \{ (u_1, u_2, \ldots, u_n) | u_i \text{ is any scalar for all } i, \ 1 \leq i \leq n \} \)
- the set of all \( m \times n \) real matrices, 
\[ M_{m,n} = \{ A = [a_{ij}] | a_{ij} \text{ is any scalar, for all } i \text{ and } j, \text{ where } 1 \leq i \leq m, 1 \leq j \leq n \} \]
\[ A = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \]
- the set of polynomials of degree \( n \) or less, 
\[ P_n = \{ a_0 + a_1x + a_2x^2 + \ldots + a_nx^n | a_i \text{ is any scalar, for all } i, \text{ where } 1 \leq i \leq n \} \]
- the space of real-valued functions on the set of real numbers, 
\[ F(\mathbb{R}) = \{ f : \mathbb{R} \rightarrow \mathbb{R} | f \text{ is any function} \} \]

Student Learning Outcomes:
1. Knowledge of the meaning and uses of forensics, digital forensics, photo forensics
2. Understanding of and ability to apply the basic notions of linear combination, spanning set, and basis
3. Understanding of how the concepts of linear combination, spanning set, and basis are used in the Red, Green, Blue (RGB) color-scheme, the Discrete Cosine Transform (DCT), and other digital imaging processes
4. Ability to compute the Discrete Cosine Transform (DCT) for image compression
5. Knowledge of careers for mathematics and science majors in the field of forensics.

Anticipated Number of Meetings: 3-4 One-hour class periods
1. Introduction

Have you taken a selfie? Or have you ever doctored a photo of yourself so that you look taller or thinner or free of blemishes or other imperfections? With the ready availability of digital technology, we have all become photographers; and, free, online photo editing tools allow us to alter our pictures to suit our pleasure. The downside of this, however, is that when we look at photos, whether in tabloids or on the Web, we wonder: Is this a fake? For, just as you can alter pictures to make yourself look more attractive, so too politicians, advertisers, terrorists, and others with particular agendas are using digital imaging tools to manipulate photos to create false impressions. Determining whether a photo has been doctored is just one of the many questions that photo forensics tries to answer.

The good news is that mathematics and computer science offer powerful tools for carrying out digital photo forensics. In this module we will focus on the connection between linear algebra and photo forensics by exploring two applications: detection of explicit images through use of the RGB color model and image reconstruction from compressed data files using the Inverse Discrete Cosine Transform. Before going into detail about these applications, this module uncovers what you know about forensics, looks at the range of issues associated with using photos as evidence in a court of law, and provides basic information about digital images.

1.1 Exploratory Questions

Answer the following questions about based on your current understanding of forensics and photo evidence.

What is forensics?

What is digital forensics?

What is photo forensics?

Where is photo evidence found?

Once you have completed these questions, look at the associated references at the end of the module and see if you would change any of your answers.

1.2 Photos as Evidence

Photographs have been used as evidence in court cases since the 1800’s. As the photograph has evolved from the first daguerreotypes to the digital images most commonly used today, so has the complexity of issues related to their use as evidence. Some of the concerns regarding photos as evidence are:

- **Finding photographic evidence:** Some photo evidence is simply provided by law enforcement who took photos of a crime scene. Defendants and plaintiffs may submit photos in the hopes of strengthening their case. An image may be used from a security camera or a bystander’s phone. Images may need to be deblurred in order to discover important details. Investigations may lead to photo evidence retrieved from computers or other digital devices. Additionally, photos may need to be recovered and reconstructed from disconnected fragments found from deleted image files.

- **Organizing and searching the photographic evidence:** Large collections of digital photographs may need to be further analyzed in search of incriminating evidence, such as explicit images, which may or may not be embedded within other images.
• **Attributing a photograph:** While it may be difficult to determine if a specific person actually took the photo, most digital images contain embedded header information that may provide information on the time the photo was taken, the device used, and the owner of the device. If header information is insufficient to attribute the photograph to a device, more complicated techniques are necessary to analyze the image in order to identify and match it to the source camera.

• **Authenticating a photograph:** In order to use a photo as evidence, it must be authenticated as being real versus a forged or synthetic image. It must also be analyzed to determine whether the image itself or header information has been altered.

2. **Digital Imaging**

When a digital camera takes a picture, it converts the actual image we see, the target image, into a matrix of tiny picture units called pixels. This matrix is called the digital image of the target image. “A digital image is a two-dimensional array of pixels. Each pixel has an intensity value (represented by a digital number) and a location address (referenced by its row and column numbers).”

For simplicity sake, let’s consider black and white images. They are produced when the brightest parts of a scene are represented by 1 (1 = white) and the darkest parts are represented by 0 (0 = black). For an example of a black and white digital image, see the 8 x 8 matrix which represents, perhaps, a face.

![Figure 1. 8 x 8 black and white image](http://logicalzero.com/gamby/reference/image_formats)

Now, the greater resolution you desire for your photo images, the larger the dimensions of the digital image matrix need to be. In the figure below you see four graphic images of the same car, all the same size but at different pixel resolutions. From left-to-right, the resolutions are: 100 pixels per square inch (PPI), 50 PPI, 25 PPI, and 10 PPI.

![Figure 2. Digital image at pixel resolutions, from left-to-right, of 100, 50, 25, and 10 PPI](http://www.finerimage.com.au/Articles/Photoshop/PixelResolution.php) [Color changed from blue to black.]
2.1 Superposition/Decomposition of a digital image into simpler components

As we noted at the beginning of this module, a major concern in photo forensics is whether a given digital image is real or fake. Do you want a picture of yourself with the President of the United States? Anyone with PhotoShop can now take a picture of themselves and superimpose that image onto an existing digital photo of the President to create the desired composite. Fake photos have become so prevalent that a number of them have risen to iconic stature such as the following photo which created the false impression that John Kerry appeared with Jane Fonda at a Vietnam War protest:

![Fake photo created by merging two separate photos](www.snopes.com/photos/politics/kerry2.asp)

So, superposition can be misused to doctor photos and create false impressions. On the other hand, the concept of superposition is a very important and a productive tool in developing photo images.

To see this, let’s focus for a moment on the set of all 2 x 2 digital black and white images and pay particular attention to these four simple matrices:

![Matrices E1, E2, E3, and E4](image)

Each of the matrices above can be considered as a 2 x 2 pixel image.

**Activity 2.1**  Copy each of the matrices $E_1, E_2, E_3,$ and $E_4$ onto its own transparency sheet.

a. Of the matrices, $E_1, E_2, E_3,$ and $E_4,$ which ones must you superimpose to create the following matrix?

![Target matrix](image)
b. Of the matrices, $E_1, E_2, E_3$, and $E_4$, which ones must you superimpose to create the following matrix?

![Matrix Image]

**Answer:** a. $E_2, E_3$  

**b. $E_1, E_3, E_4$**

**Weighted Sums**

Suppose that in your Linear Algebra course, your final grade will be based on 3 exams ($E_1, E_2, E_3$) each counting 20%, the final exam ($E_4$) which counts 20%, homework ($E_5$) which counts 10%, and a project ($E_6$) which counts 10%. Then your final grade is computed as follows:  

$$\text{GRADE} = .20E_1 + .20E_2 + .20E_3 + .20E_4 + .10E_5 + .10E_6.$$  

Just as your final grade in a course,  

$$\text{GRADE} = w_1E_1 + w_2E_2 + w_3E_3 + \ldots + w_nE_n$$  

is a weighted sum of your grades $E_1, E_2, E_3$, and $E_n$ corresponding to the $n$ pieces of required coursework, so, too, any $2 \times 2$ black and white digital image, $M$, can be expressed as a weighted sum of the above matrices $E_1, E_2, E_3$, and $E_4$.

For example, the $2 \times 2$ black and white digital image below, $M$, can be expressed as  

$$M = 1E_1 + 1E_2 + 0E_3 + 0E_4$$

![Image of Black and White Matrix]

where $+$ means “superimpose with.” So for any matrix $E_i$, $1E_i$ means use $E_i$ in the superposition and $0E_i$ means not to use $E_i$ in the superposition.
Activity 2.2  For 2 x 2 black and white digital images,
a. List all possible black and white digital images \( M \) as a weighted sum of the four matrices \( E_1, E_2, E_3, \) and \( E_4 \) as follows:

\[
M = w_1E_1 + w_2E_2 + w_3E_3 + w_4E_4
\]

Note: In the digital image context, a weight is either 0 or 1. The weights do not need to sum to 1 as with the GRADE weighted sum.

b. Is there any 2 x 2 black and white digital image which can be expressed in more than one way?

Answer:
a. Sixteen possible images:

<table>
<thead>
<tr>
<th>( M_1 )</th>
<th>( M_2 )</th>
<th>( M_3 )</th>
<th>( M_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 0E_1 + 0E_2 + 0E_3 + 0E_4 )</td>
<td>( 1E_1 + 0E_2 + 0E_3 + 0E_4 )</td>
<td>( 0E_1 + 1E_2 + 0E_3 + 0E_4 )</td>
<td>( 0E_1 + 0E_2 + 1E_3 + 0E_4 )</td>
</tr>
<tr>
<td>( 0E_1 + 1E_2 + 0E_3 + 0E_4 )</td>
<td>( 1E_1 + 0E_2 + 0E_3 + 0E_4 )</td>
<td>( 0E_1 + 1E_2 + 0E_3 + 0E_4 )</td>
<td>( 0E_1 + 0E_2 + 1E_3 + 0E_4 )</td>
</tr>
<tr>
<td>( 1E_1 + 0E_2 + 1E_3 + 0E_4 )</td>
<td>( 0E_1 + 1E_2 + 0E_3 + 0E_4 )</td>
<td>( 1E_1 + 0E_2 + 0E_3 + 1E_4 )</td>
<td>( 0E_1 + 1E_2 + 0E_3 + 1E_4 )</td>
</tr>
<tr>
<td>( 0E_1 + 1E_2 + 1E_3 + 0E_4 )</td>
<td>( 1E_1 + 0E_2 + 0E_3 + 1E_4 )</td>
<td>( 0E_1 + 1E_2 + 1E_3 + 0E_4 )</td>
<td>( 1E_1 + 0E_2 + 1E_3 + 0E_4 )</td>
</tr>
<tr>
<td>( 1E_1 + 1E_2 + 0E_3 + 1E_4 )</td>
<td>( 1E_1 + 0E_2 + 1E_3 + 1E_4 )</td>
<td>( 1E_1 + 1E_2 + 1E_3 + 0E_4 )</td>
<td>( 1E_1 + 1E_2 + 1E_3 + 1E_4 )</td>
</tr>
</tbody>
</table>

b. No.

Activity 2.3  In this activity you will extend your thinking in the last exercise to 8 x 8 images. Let \( E_{i,j} \) be the 8 x 8 matrix whose \( (i,j) \) entry is solid black, with the other entries all white. For example, \( E_{2,3} \) is given by

Express the letter A, shown as an 8 x 8 black and white image below, as a weighted sum of the matrices \( E_{i,j} \). You do not need to write the matrices which have a zero weight.
Answer:
$E_{1,4} + E_{1,5} + E_{2,3} + E_{2,6} + E_{3,2} + E_{3,7} + E_{4,2} + E_{4,7} + E_{5,2} + E_{5,3} + E_{5,4} + E_{5,5} + E_{5,6} + E_{5,7} + E_{6,2} + E_{6,7} + E_{7,2} + E_{7,7}$

In the previous examples, any M x N black and white digital image can be created by superposition of associated M x N matrices each with only one pixel colored black. Later we will extend the idea of superposition to overlay colors with different weights onto a single pixel in order to create a new color.

### 2.2 Forensics Application: Human Body Image Detection

One primary area of research in digital forensics is the need to search and categorize large sets of image data for explicit images. Internet and email providers and search engines use filters to block unwanted explicit images from their customers. Many companies use explicit image detection software that may implicate employees in using company computers and time for inappropriate content. Law enforcement searches computers for illegal digital contraband such as child pornography.

There are many factors that must be considered for automated explicit image detection. The key component is skin color analysis and the ability to accurately detect skin regions. The more a program tries to detect body parts or contours, the more computationally expensive it becomes. The software may also have difficulty distinguishing between a benign photo of a family playing at the beach from sexually explicit images. Additional algorithms are also needed to distinguish adult faces from children. Despite these limitations and the fact that the definition of what is explicit may be subjective, successful software will find the unwanted content and minimize an individual’s exposure to it.

Since skin color detection is one of the most important aspects of human body detection, the techniques rely heavily on color models.

#### 2.2.1 The RGB Color Model

Our modern understanding of color and light can be traced back to Sir Isaac Newton and results published in 1672. He conducted experiments in which he observed that a triangular glass prism can separate white light into seven rainbow colors: red, orange, yellow, green, blue, indigo, and violet. From this group of colors, the entire spectrum of colors could be created. Later it was determined
that only three colors were needed to create the other colors. While there are many color models that are used, this module will focus on the simple RGB (Red, Green, Blue) color model.

When three beams of light of equal intensity in these primary colors are projected on a screen, the overlapping regions produce new secondary colors similar to those shown in Figure 4. Red and green produce yellow, red and blue produce magenta, and green and blue produce cyan. The region in the center, where all three beams of light overlap, results in white. The absence of the three colors is black. By varying the intensities of the light beams a broad range of colors can be produced.

![RGB additive color model](http://en.wikipedia.org/wiki/RGB_color_model)

Figure 4. RGB additive color model

The RGB color model is additive in that the wavelengths of the different color intensities are added together to produce the new color. Similarly RGB models are used in digital images where each pixel represents a single color produced by adding various intensities of red, green, and blue. Typically the intensities of each color have numerical values that range from 0 to 255 corresponding to a binary 8-bit byte of information \(2^8 = 256\). This results in over 16.7 million possible color combinations. This leads us back to the digital forensics question of determining which of these millions of color combinations are possible skin tones.

This following activities will use a java applet found on the Demos with Positive Impact site to explore the RGB color model. Go to the following website to access the RGB java applet [www.mathdemos.org/mathdemos/RGBJAVA/](http://www.mathdemos.org/mathdemos/RGBJAVA/). You should see a screen similar to the one below.

*For a different hands-on activity, this activity may be modified to use Red, Blue, and Green food coloring in milk. Rather than intensity values between 0 and 1, you must count the number of food coloring droplets. However, the food coloring is not as exact as the pixel color intensity and you may get different color results. Then a linear combination can be described using the number of droplets as the weights.

†If you receive an error when loading the applet, you may need to update Java and/or add the website to the Java Exception Site List.
You can use the sliders to adjust the intensities of each of the Red, Green, and Blue colors to produce new colors that will appear in the large square to the right of the sliders. The intensity of each color ranges between 0 and 1 in increments of 0.005, resulting in 201 possible intensities for each color. Hence, this applet has $201^3 = 8,120,601$ color combinations.

**Activity 2.4** Verify the following basic color combinations.

a. Red = 1.0 Red + 0.0 Green + 0.0 Blue; Green = 0.0 Red + 1.0 Green + 0.0 Blue; Blue = 0.0 Red + 0.0 Green + 1.0 Blue

b. Black = 0.0 Red + 0.0 Green + 0.0 Blue; White = 1.0 Red + 1.0 Green + 1.0 Blue

c. Yellow = 1.0 Red + 1.0 Green + 0.0 Blue; Magenta = 1.0 Red + 0.0 Green + 1.0 Blue; Cyan = 0.0 Red + 1.0 Green + 1.0 Blue

**Answer:** Given above – verification only.

**Activity 2.5** Explore same intensity colors.

a. Before using the applet, give a conjecture about what color you might expect to see if all three intensities are set to 0.50: _______________.

Now use the applet to determine whether your conjecture was correct:

0.50 Red + 0.50 Green + 0.50 Blue = _______________

b. Choose an intensity and set all three colors to the same intensity value. Repeat this process with different intensity values until you can determine the behavior of the resulting color. Explain your observation in a few brief sentences below.

**Answer:**

a. Conjecture answers may vary; Actual answer is grey.

b. Black is obtained by all intensities equal to 0 and white is obtained by all intensities equal to 1. For other intensity values all equal, the resulting colors are varying shades of grey.

**Activity 2.6** Name the color given by each of the combinations below.

a. 0.70 Red + 0.20 Green + 0.80 Blue = _______________

b. 0.75 Red + 0.90 Green + 1.00 Blue = _______________

**Answer:**

a. Purple  

b. Light Blue
**Activity 2.7**  
Determine the intensities to create the following colors.

a. Pink = ______ Red + ______ Green + ______ Blue  
b. Orange = ______ Red + ______ Green + ______ Blue

**Answer:** Answers may vary. One set of possible answers is given below.  
a. Pink = 0.90 Red + 0.50 Green + 0.80 Blue  
b. Orange = 0.87 Red + 0.60 Green + 0.26 Blue

As seen in the activity with the applet, Color can be expressed in two ways:

\[
\text{Color} = a(\text{Red}) + b(\text{Green}) + c(\text{Blue})
\]

AND

\[
\text{Color} = a(1,0,0) + b(0,1,0) + c(0,0,1)
\]

where a, b, and c the intensities of red, green, and blue, respectively, and 0 ≤ a, b, c ≤ 1.

This second expression shows how each color can be represented by a point in three-dimensional space. For example, since Yellow = 1.0(\text{Red}) + 1.0(\text{Green})+0.0(\text{Blue}), then Yellow can also be represented by the point (1,1,0) because: Yellow = 1(1, 0, 0) + 1(0, 1, 0) +0(0, 0, 1) = (1, 1, 0).

The figure below shows this graphical representation for the RGB color palette.

![RGB Color Palette](http://wingsforall.com/bhagouauty/blog/punjabi-the-gurmukhi-languages-as-the-expansion-of-gaytri-mantra-eak-oankar/)

**Figure 6.** The RGB color model mapped to a cube.

We conclude this section with a couple of notes about color and digital images.

First, with black and white images, each pixel is colored either black (0 = black) or white (1 = white). To produce images on a gray-scale, each pixel has an intensity value between 0 and 1; in the gray-scale, for example, a pixel with intensity value .2 is darker than a pixel with intensity value .6. With color images, on the other hand, each pixel is assigned three numbers, each between 0 and 1, which represent the intensity values of Red, Green, and Blue, respectively. The following two graphics, produced with MATLAB, illustrate how individual pixels are assigned numbers in the gray-scale color model and the RGB color model:
Last note, there is no universal RGB model. Rather, “RGB is a device-dependent color model: different devices detect or reproduce a given RGB value differently, since the color elements (such as phosphors or dyes) and their response to the individual R, G, and B levels vary from manufacturer to manufacturer, or even in the same device over time.” Therefore, one aspect of photo image forensics is to use this RGB variation to determine whether a particular camera was used to produce a particular photo image.

**Section 2 Homework**

**Exercise 2.1** Use the RGB java applet [www.mathdemos.org/mathdemos/RGBJAVA/](www.mathdemos.org/mathdemos/RGBJAVA/) to explore skin tones in the RGB model by answering the questions below.

a. Use the sliders to adjust the RGB intensities until you find a color that you think represents a good skin tone color. (You may use your own skin color as a guide). Write down the intensities you found. 
*Skin Tone =   ______ Red + ______ Green + ______ Blue.*

b. Fix the Green and Blue intensities to be the values found in part (a). Now adjust the Red intensity to find upper and lower limits of where you think the color still looks like a skin tone. 
*Green intensity:   _____  Blue intensity:  _____  Range of Red intensities:  ______

Determine how many different skin tone colors you found by subtracting the upper and lower limits, then dividing by 0.005. 
*Number of skin tone colors found: ____________

c. Fix the Red and Green intensities to be the values found in part (a). Now adjust the Blue intensity to find upper and lower limits of where you think the color still looks like a skin tone. 
*Red intensity:   ______  Green intensity:  _____  Range of Blue intensities:  _____

Determine how many different skin tone colors you found by subtracting the
upper and lower limits, then dividing by 0.005.
Number of skin tone colors found: ____________

d. Fix the Red and Blue intensities to be the values found in part (a). Now adjust the Green intensity to find upper and lower limits of where you think the color still looks like a skin tone.
Red intensity: _____ Blue intensity: _____ Range of Green intensities: ______

Determine how many different skin tone colors you found by subtracting the upper and lower limits, then dividing by 0.005.
Number of skin tone colors found: ____________

e. Based on your exploration in parts (a)-(d), briefly describe why it is difficult for a computer to detect photos of humans based on skin color only. What other problems might a computer program encounter?

Answer: All answers will vary. The point of this exercise is to demonstrate that hundreds or thousands of different intensity combinations will result in colors that could be a skin tone. One set of possible answers is given below.

a. Skin Tone = 0.700 Red + 0.500 Green + 0.300 Blue
b. Green intensity: 0.500 Blue intensity: 0.300 Range of Red intensities: 0.525-0.800
   Number of skin tone colors found = (0.800 − 0.525)/.005 = 55
c. Red intensity: 0.700Green intensity: 0.500 Range of Blue intensities: 0.155-0.460
   Number of skin tone colors found = (0.460 − 0.155)/.005 = 61
d. Red intensity: 0.700Blue intensity: 0.300 Range of Green intensities: 0.420-0.610
   Number of skin tone colors found = (0.460 − 0.155)/.005 = 38
e. Answers will vary, but here are some possibilities. Since there are hundreds or thousands of intensity combinations that result in possible skin tones, a computer would have to check all of them to find an image of a human. Other complications that students may consider: A computer may detect animals and other objects that have human skin coloring. If the digital images are in black and white, then skin tone detection may not be possible.

3. Fundamental concepts: linear combination, spanning set, basis

In the color exploration activities above, we saw that every color in the spectrum can be expressed as a weighted sum of the three primary colors Red, Green, and Blue; moreover, for each color, there is one and only one way to express it as a weighted sum of Red, Green, and Blue.

It is important to note that the set of all colors, \( \{a \text{Red} + b \text{Green} + c \text{Blue} \mid 0 \leq a, b, c \leq 1\} \), is NOT a vector space because scalar multiplication of Red, Green, and Blue is neither defined for negative scalars nor for positive scalars greater than 1. (Equivalently, the unit cube shown in Figure 6 is not a subspace of \( \mathbb{R}^3 \).)
On the other hand, because every color in the spectrum is a sum of scalar multiples of Red, Green, and Blue, we say that every color is a linear combination of the three colors Red, Green, and Blue. In addition, we call the set \( S = \{\text{Red, Green, Blue}\} \) a spanning set for the set of all colors.

### 3.1 Terminology and definitions
We now present the formal, vector space definitions for spanning set and linear combination.

**Definition**
Let \( V \) be any vector space and let \( S = \{u_1, u_2, \ldots, u_k\} \) be a set of vectors in \( V \).

If every vector \( v \) in \( V \) can be expressed as
\[
v = c_1u_1 + c_2u_2 + \cdots + c_ku_k,
\]
for some scalars \( c_1, c_2, \ldots, c_k \), then \( S \) is called a spanning set for \( V \) and we say that every vector \( v \) in \( V \) is a linear combination of the vectors in set \( S \).

**Example 3.1**
In vector space \( \mathbb{R}^2 \), every vector \( v = (a, b) \) can be expressed as follows:
\[
v = (a, b) = a(1, 0) + b(0, 1)
\]
where \( a \) and \( b \) are any scalars. If we let \( u_1 = (1,0) \), \( u_2 = (0,1) \), and \( S = \{u_1, u_2\} \), then we see that \( v \) is a linear combination of \( u_1 \) and \( u_2 \). So, \( S = \{u_1, u_2\} \) is a spanning set for \( \mathbb{R}^2 \).

Likewise in \( \mathbb{R}^3 \), the set \( S = \{(1,0,0), (0,1,0), (0,0,1)\} \) is a spanning set for \( \mathbb{R}^3 \).
This is because any vector \( v = (a, b, c) \) in \( \mathbb{R}^3 \) may be written as a linear combination of the vectors in \( S \) as follows:
\[
v = (a, b, c) = a(1,0,0) + b(0,1,0) + c(0,0,1)
\]

**Example 3.2**
The set \( S = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \right\} \) is a spanning set for the vector space of \( 2 \times 2 \) matrices, \( M_{2,2} \). This is because any matrix \( A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \) in \( M_{2,2} \) can be expressed as a linear combination of the members of \( S \) as follows:
\[
A = a_{11} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + a_{12} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + a_{21} \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + a_{22} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}
\]

**Example 3.3**
In the vector space \( P_n \) of all polynomials of degree \( n \) or less, the set
\( S = \{1, x, x^2, \ldots, x^n\} \) is a spanning set for \( P_n \) because each of its members,
\[
a_0 + a_1x + a_2x^2 + \cdots + a_nx^n,
\]
where \( a_i \) is any scalar for \( 1 \leq i \leq n \), is a linear combination of the members of \( S \).

**Activity 3.1**
Spanning sets and linear combinations
a. Show that \( S = \{(1,1),(2,1)\} \) is a spanning set for \( \mathbb{R}^2 \).
b. Show that \( T = \{1, 1 + x, 1 + x^2\} \) is a spanning set for \( P_2 \).

c. Find a spanning set for each of the following vector spaces of \( 2 \times 2 \) matrices:

i. \( W = \left\{ \begin{bmatrix} a & b \\ 0 & 0 \end{bmatrix} : a, b \text{ are any real numbers} \right\} \)

ii. \( W = \left\{ \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix} : a \text{ is any real number} \right\} \)

d. Find a spanning set for the set of \( 2 \times 2 \) black and white digital images.

**Answer:**

a. Let \((a, b)\) be in \( \mathbb{R}^2 \). We must find real numbers \( c \) and \( d \) such that \((a, b) = c(1,1) + d(2,1)\); that is, we must find \( c \) and \( d \) in \( \mathbb{R} \) such that

\[
(1) \quad a = c + 2d \\
(2) \quad b = c + d .
\]

Now, subtracting (2) from (1) gives: \( a - b = d \); further, subtracting (1) from \( 2 \times (2) \) gives: \( 2b - a = c \).

Therefore, there exists \( c \) and \( d \) in \( \mathbb{R} \), namely \( c = 2b - a \) and \( d = a - b \), such that \((a, b) = c(1,1) + d(2,1)\).

So, \( S = \{(1,1), (2,1)\} \) is a spanning set for \( \mathbb{R}^2 \).

b. Let \( a_0 + a_1 x + a_2 x^2 \) be in \( P_2 \). We must find real numbers \( c, d, \) and \( e \) such that

\[
a_0 + a_1 x + a_2 x^2 = c(1) + d(1 + x) + e(1 + x^2) ;
\]

that is, we must find \( c, d, \) and \( e \) in \( \mathbb{R} \) such that

\[
(1) \quad a_0 = c + d + e \\
(2) \quad a_1 = d \\
(3) \quad a_2 = e .
\]

Subtracting (2) + (3) from (1) gives: \( c = a_0 - a_1 - a_2 \).

Therefore, there exists \( c, d, \) and \( e \) in \( \mathbb{R} \), namely \( c = a_0 - a_1 - a_2, \; d = a_1, \) and \( e = a_2 \), such that \( a_0 + a_1 x + a_2 x^2 = c(1) + d(1 + x) + e(1 + x^2) \).

Therefore, \( T = \{1, 1 + x, 1 + x^2\} \) is a spanning set for \( P_2 \)

c. i. \( \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right\} \)

ii. \( \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right\} \)

d. \[\text{Diagrams} \]
Another fundamental linear algebra concept that is needed for the compression and decompression stages of digital imaging processing is the concept of a \textit{basis}.

\textbf{Definition} \hspace{1em} Let $V$ be any vector space and let $S = \{v_1, v_2, \ldots, v_k\}$ be a spanning set for $V$. Then $S$ is called a \textbf{basis} for $V$ if there is one and only one way to express each vector $v$ in $V$ as a linear combination of the vectors in $S$.

\textbf{Activity 3.2} \hspace{1em} Let $V$ be any vector space and let $S = \{v_1, v_2, \ldots, v_k\}$ be a spanning set for $V$. Show that $S$ is a basis for $V$ if and only if whenever $0 = c_1 v_1 + c_2 v_2 + \ldots + c_k v_k$ then $c_i = 0$ for $i = 1, 2, \ldots, k$.

\textbf{Answer:} \\
$\Rightarrow$: Suppose that $S = \{v_1, v_2, \ldots, v_k\}$ is a spanning set for a vector space $V$ and that $S$ is a basis for $V$. Since the zero vector is in $V$, it can be expressed as $0 = c_1 v_1 + c_2 v_2 + \ldots + c_k v_k$. The zero vector can also always be expressed as $0 = 0v_1 + 0v_2 + \ldots + 0v_k$. Since $S$ is a basis, the linear combination for the zero vector is unique. Therefore $c_i = 0$, for $i = 1, 2, \ldots, k$.

$\Leftarrow$: On the other hand, suppose that $S = \{v_1, v_2, \ldots, v_k\}$ is a spanning set for a vector space $V$ and that $S$ satisfies the following condition:

whenever $0 = c_1 v_1 + c_2 v_2 + \ldots + c_k v_k$, then $c_i = 0$, for $i = 1, 2, \ldots, k$ \hspace{1em} (*)

If a member $v$ in $V$ can be expressed in two ways as

1. $v = c_1 v_1 + c_2 v_2 + \ldots + c_k v_k$ and
2. $v = d_1 v_1 + d_2 v_2 + \ldots + d_k v_k$

then subtracting (2) from (1) we get: $0 = (c_1 - d_1) v_1 + (c_2 - d_2) v_2 + \ldots + (c_k - d_k) v_k$

By (*), it follows that $c_i - d_i = 0$, for $i = 1, 2, \ldots, k$; that is, $c_i = d_i$ for $i = 1, 2, \ldots, k$. Therefore, there is one and only one way to express each vector in $V$ as a linear combination of the vectors in $S$. Therefore, $S$ is a basis for $V$.

\textbf{Note} \hspace{1em} As we noted earlier, the set of all colors in the spectrum is not a vector space because, for example, multiplication of Red, Green, or Blue by negative scalars is not defined. On the other hand, the set $\{\text{Red}, \text{Green}, \text{Blue}\}$ is a spanning set for the set of all colors; furthermore, every color in the spectrum has a unique representation as a linear combination of Red, Green, and Blue.

\textbf{Note} \hspace{1em} The set of $2 \times 2$ black and white digital images is also not a vector space because scalar multiplication of images is only defined for 0 and 1. On the other hand, the set $\{E_1, E_2, E_3, E_4\}$ is a spanning set for the set of all $2 \times 2$ black and white images; furthermore,
every 2 x2 black and white image has a unique representation as a linear combination of $E_1, E_2, E_3$, and $E_4$.

### 3.2 Forensics Application: Image Reconstruction and Deblurring

A target image that you view can be considered as a continuous image in the spatial domain. When you use a camera to take a picture of the target image, the resulting digital image is only a discrete approximation to that continuous image. The target image is sampled at discrete points which correspond to the pixels in the digital image. Each pixel then contains the information regarding the intensities of Red, Blue, and Green at each sample point. If the pixel resolution is high enough, our eye will perceive the picture as a continuous image. If there are not enough pixels, the image will look choppy as the borders between pixels can be detected by the human eye.

While a picture may look better with more pixels, it will require much more computer space to store and computer time to process or transmit. Hence much work has gone into determining ways to compress digital images so that they take up less storage space. There are two general types of image compression algorithms, lossless and lossy. A **lossless algorithm** takes advantage of redundancies in the data to compress and store the data, but the decompressed image will contain all of the original information. A **lossy algorithm** also takes advantage of redundancies but loses some information because it removes or averages the redundancies. So when an image is decompressed, it is only an approximation to the original image as it will not contain all of the original information.

One of the most commonly used compression algorithms is the JPEG standard, which is a lossy method. The fundamental process for JPEG compression is given in Figure 8 and described below.

![Figure 8. JPEG Compression Process.](image)

First, the original image data is divided into 8 x 8 pixel blocks. The Discrete Cosine Transform (DCT) is applied to each block, resulting in coefficients (or weights) that will be used to reconstruct the image by superposition of functions during the reverse process. The DCT and its inverse will be discussed in more detail in later sections. After the data is transformed, it undergoes a process called quantization which is basically a method of reducing the magnitude of the coefficients and rounding them. Hence, this is the lossy part of the algorithm since some detail is lost by rounding. The details of quantization will be left for the student as a project. Finally, the quantized data is encoded, often with Huffman codes, for efficiency. This module does not discuss the details of entropy encoding.

Once the image is compressed, the approximation to the original can be retrieved by using the inverse of the steps in the reverse order.

Understanding the JPEG process is important in digital image forensics. For example, the quantization tables and the encoding information are often stored with the compressed image data so that both can be used for decompressing the image properly. Since different manufacturers use different tables, this information can provide forensics investigators information about which
camera produced the digital image. Also, if an image was compressed, decompressed, and then tampered with before compressing again, it would mean that the approximated image data was being quantized the second time. Since this lossy part of the algorithm would be applied twice, it could result in abnormalities in the compressed image data and provide evidence of the tampering.¹⁰

Law enforcement use digital images to help them in their investigations. They may have photos taken from a phone or surveillance camera, which are often blurry. Algorithms aimed at sharpening blurry images rely on understanding the cause of the blur, such as lossy JPEG compression. Two other common reasons for blurry pictures are being out-of-focus or containing motion blur where either the image being captured was in motion (e.g., a person running) or the camera was not held still.¹¹

The remainder of this module is aimed at understanding the Discrete Cosine Transform (DCT) and its inverse (IDCT) as it relates to functions and images. Before getting to the transforms, we will start by discussing the superposition of functions.

### 3.3 Superposition of Functions

As we saw earlier in this module, any 2x2 black and white pixel image is a linear combination of four simple 2x2 black and white images E₁, E₂, E₃, and E₄, the set of which is a spanning set for the set of all 2x2 black and white images. We also saw that a linear combination in the context of images means superposition of E₁, E₂, E₃, and E₄.

We now take a look at the concepts of spanning set, linear combination, basis, and superposition in the context of graphing functions. Photo images can be realized as linear combinations of basic sets of functions, so we want to review how to add functions graphically as well as how to graph scalar multiples of a function when presented with its graph. By the superposition of two functions, f and g, in \( F(\mathbb{R}) = \{ f : \mathbb{R} \rightarrow \mathbb{R} \mid f \text{ is any function} \} \), we will mean the function \( f + g \) that is obtained from ordinate addition of \( f \) and \( g \).

**Activity 3.3** *Geometry activity:* Consider the following graphs for two functions \( y = f(x) \) and \( y = g(x) \):
Using the above graphs, graph each of the following functions:

a. \( y = 2f \)
b. \( y = -g \)
c. \( y = f + g \)
d. \( y = 2f - g \)

Answer:

3.4 One-Dimensional Discrete Cosine Transform

The concepts of linear combinations and basis are the essential building blocks for the Discrete Cosine Transform (DCT), a widely used algorithm in digital image compression.

Before looking at the 2-D DCT for 2-D images, we will consider the simpler 1-D DCT. In this case, an image can be considered as a one-dimensional spatial function \( f(x) \). The continuous function is sampled at \( N \) discrete points to get the values of \( f \) at \( x = 0, 1, \ldots, N - 1 \). Then for \( f(x) \) and \( N \), the 1-D DCT is defined as

\[
C(i) = a(i) \sum_{x=0}^{N-1} f(x) \cos \left( \frac{(2x+1)i\pi}{2N} \right) \quad \text{for } i = 0, 1, \ldots, N - 1
\]

where

\[
a(i) = \begin{cases} 
\sqrt{\frac{1}{N}} & \text{if } i = 0 \\
\sqrt{\frac{2}{N}} & \text{otherwise}
\end{cases}
\]

So given a function \( f(x) \), the DCT coefficients \( C(i) \) are found by using the discrete values of the function \( f(x) \) at the points where \( x = 0, 1, \ldots, N - 1 \). Hence the DCT takes a discrete set of \( N \) points in the spatial domain \( x \) and transforms them into a new set of \( N \) points in the frequency domain \( i \). Note that all \( N \) values of \( f(x) \) contribute to each coefficient \( C(i) \).
On the other hand, given a set of DCT coefficients, \( C(i), i = 0, 1, \ldots N - 1 \), the discrete function \( f(x) \) is recovered by applying the 1-D Inverse Cosine Transform (IDCT):

\[
f(x) = \sum_{i=0}^{N-1} C(i) a(i) \cos \left( \frac{(2x + 1)i\pi}{2N} \right) \quad \text{for } x = 0, 1, \ldots N - 1
\]

where the \( a(i) \) are the same as given above. It is here, in the inverse transform, that one can see that the function \( f(x) \) is a linear combination of the basis functions \( a(i) \cos [(2x+1)i\pi/2N] \) with the weights given by the DCT coefficients \( C(i) \).

Figure 9 below shows the basis functions for \( N = 8 \) where the graphs on the left are continuous versions of the discrete functions graphed on the right.

\[\text{Figure 9.} \quad \text{Graphs of the discrete basis functions for } N = 8 \text{ are shown on the right (vertical lines are added to aid in visualization). The continuous counterparts are given on the left.}\]
Example 3.4  Given the function \( f(x) = \begin{cases} x & \text{for } 0 \leq x \leq 3 \\ 3 & \text{for } 3 < x \leq 7 \end{cases} \), compute the DCT coefficients (for \( N = 8 \)).

The graph for \( f(x) \) is shown below.

\[
C(0) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \cos \left[ \frac{(2x+1)(0)\pi}{16} \right] = \sqrt{\frac{1}{8}} \sum_{x=0}^{7} f(x)
= \sqrt{\frac{1}{8}} \left[ f(0) + f(1) + f(2) + f(3) + f(4) + f(5) + f(6) + f(7) \right]
= \sqrt{\frac{1}{8}} \left[ 0 + 1 + 2 + 3 + 3 + 3 + 3 + 3 \right]
\approx 6.364
\]

\[
C(1) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \cos \left[ \frac{(2x+1)(1)\pi}{16} \right] = \sqrt{\frac{1}{8}} \sum_{x=0}^{7} f(x) \cos \left[ \frac{(2x+1)\pi}{16} \right]
= \sqrt{\frac{1}{4}} \left[ f(0) \cos \left( \frac{\pi}{16} \right) + f(1) \cos \left( \frac{3\pi}{16} \right) + f(2) \cos \left( \frac{5\pi}{16} \right) + f(3) \cos \left( \frac{7\pi}{16} \right) + f(4) \cos \left( \frac{9\pi}{16} \right) + f(5) \cos \left( \frac{11\pi}{16} \right) + f(6) \cos \left( \frac{13\pi}{16} \right) + f(7) \cos \left( \frac{15\pi}{16} \right) \right]
= \frac{1}{2} \left[ 0 \cdot \cos \left( \frac{\pi}{16} \right) + 1 \cdot \cos \left( \frac{3\pi}{16} \right) + 2 \cdot \cos \left( \frac{5\pi}{16} \right) + 3 \cdot \cos \left( \frac{7\pi}{16} \right) + 3 \cdot \cos \left( \frac{9\pi}{16} \right) + 3 \cdot \cos \left( \frac{11\pi}{16} \right) + 3 \cdot \cos \left( \frac{13\pi}{16} \right) + 3 \cdot \cos \left( \frac{15\pi}{16} \right) \right]
\approx -2.580
\]

The remaining coefficients are found in the same way. The results are summarized in the table below.

<table>
<thead>
<tr>
<th>Spatial Domain</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f(x) )</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Frequency Domain</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C(i) )</td>
<td>6.364</td>
<td>-2.580</td>
<td>-1.577</td>
<td>-0.562</td>
<td>0.000</td>
<td>0.050</td>
<td>-0.112</td>
<td>-0.153</td>
</tr>
</tbody>
</table>
3.5 Two-Dimensional Discrete Cosine Transform

Given a two-dimensional image, the image can be thought of as a two-dimensional spatial function \( f(x,y) \). As mentioned before, the image is sampled at discrete points to create the discrete digital image. The sample points correspond to the pixels in the digital image and \( f(x,y) \) gives the pixel value at the point \((x,y)\). The digital image is then divided into \( M \times N \) blocks, typically \( 8 \times 8 \), and the 2D-DCT is applied to each block. Given \( f \), \( M = 8 \), and \( N = 8 \), the 2D-DCT is defined as

\[
C(i,j) = a(i)a(j) \sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) \cos \left( \frac{(2x+1)i\pi}{16} \right) \cos \left( \frac{(2y+1)j\pi}{16} \right) \text{ for } i,j = 0,1,...,7
\]

where

\[
a(i) = a(j) = \begin{cases} 
\frac{1}{\sqrt{8}} & \text{if } i, j = 0 \\
\frac{1}{2} & \text{otherwise} 
\end{cases}
\]

The 2D-DCT takes a discrete set of \( 8 \times 8 = 64 \) points in the spatial domain \((x,y)\) and converts them to the frequency domain \((i,j)\). The resulting coefficients \( C(i,j) \) are the weights that will be used to reconstruct the image using the 2D-IDCT:

\[
f(x,y) = \sum_{i=0}^{7} \sum_{j=0}^{7} C(i,j)a(i)a(j) \cos \left( \frac{(2x+1)i\pi}{16} \right) \cos \left( \frac{(2y+1)j\pi}{16} \right) \text{ for } x,y = 0,1,...,7
\]

In the case of JPEG compression, \( M = N = 8 \) and there are 64 resulting basis functions corresponding to \( a(i)a(j) \cos \left( \frac{(2x+1)i\pi}{16} \right) \cos \left( \frac{(2y+1)j\pi}{16} \right) \text{ for } i,j = 0,1,...,7 \). Hence, the recovered image \( f(x,y) \) is a linear combination of these functions with weights \( C(i,j) \). These 64 basis functions are the discrete patterns given in Figure 10 below.

![Figure 10. Graphical representation of the 2D-DCT for N = 8 (8 x 8 blocks) (http://en.wikipedia.org/wiki/Discrete_cosine_transform).](http://en.wikipedia.org/wiki/Discrete_cosine_transform)
Since one of the primary foci of this module is to have a visual understanding of the idea of superposition and linear combinations, we leave the details of computing the 2D-DCT coefficients and the Inverse DCT for further investigation by the student.

A good demonstration of how a linear combination of the basis functions is used to re-create an image can be found at http://en.wikipedia.org/wiki/Discrete_cosine_transform under the heading Example 3.5 Example of IDCT. Start with an 8 x 8 pixel image of the capital letter A as shown in Figure 11.

Figure 11. The original letter A as an 8 x 8 pixel image on the left. It is magnified by 10 in the middle and right images (http://en.wikipedia.org/wiki/Discrete_cosine_transform).

The 2D-DCT coefficients are computed for the image using the equations above and given in the matrix shown in Figure 12:

![Figure 12](image)

Figure 12. The 2D-DCT coefficients for the image in Figure 11. The circled values correspond to the coefficients shown in Figure 13.

The final image of the letter A is reconstructed as the Inverse DCT and shown in Figure below. This final image is essentially a linear combination of the basis functions with their associated DCT coefficients as the weights. Each figure contains three images: The image on the right is the DCT basis function along with the computed DCT coefficient. The middle image shows what the basis function multiplied by the associated weight looks like. The left image shows the resulting linear combination using up to 10, 36, and all 64 basis functions.

![Figure 13](image)

Figure 13. Graphical representation of the Inverse DCT using 10, 36, and all 64 basis functions. See http://en.wikipedia.org/wiki/Discrete_cosine_transform for an animation of adding in all 64 basis functions.

Section 3 Homework

Exercise 3.1 Show that \( T = \{ 1, 1 + x, 1 + x^2 \} \) is a basis for the vector space \( P_2 \).
Answer: We know that \( T = \{1, 1 + x, 1 + x^2\} \) is a spanning set for \( P_2 \) by Activity 3.1. Suppose that \( 0 + 0x + 0x^2 = c(1) + d(1 + x) + e(1 + x^2) \) for some \( c, d, \) and \( e \) in \( P_2 \). Then by equating like coefficients we get:

\[
\begin{align*}
(1) \quad 0 &= c + d + e \quad \text{and} \\
(2) \quad 0 &= d \\
(3) \quad 0 &= e.
\end{align*}
\]

By substitution for \( d \) and \( e \) into (1), it follows that \( c = 0 \). So, \( c = 0, d = 0, \) and \( e = 0 \). Therefore, by Activity 3.2, we conclude that \( T \) is a basis for \( P_2 \).

Exercise 3.2 Explain why \( S = \{1, 1 + x, 1 - x, 1 + x^2\} \) is not a basis for \( P_2 \).

Answer: First, \( S = \{1, 1 + x, 1 - x, 1 + x^2\} \) is a spanning set for \( P_2 \) since any member \( a_0 + a_1x + a_2x^2 \) in \( P_2 \) can be expressed as \( (a_0 - a_1 - a_2)1 + a_3(1 + x) + 0(1 - x) + a_4(1 + x^2) \).

On the other hand, since \( 0 + 0x + 0x^2 = -2(1) + 1(1 + x) + 1(1 - x) + 0(1 + x^2) \), then by Activity 3.2 it follows that \( S \) is not a basis for \( P_2 \).

Exercise 3.3 Given \( f(x) = \begin{cases} 
0 & \text{for } 0 \leq x \leq 1 \\
1 & \text{for } 1 < x \leq 3
\end{cases} \), compute all of the DCT coefficients for \( N = 4 \).

Answer:

<table>
<thead>
<tr>
<th>Spatial Domain ( x )</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f(x) )</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Frequency Domain ( i )</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C(i) )</td>
<td>1.000</td>
<td>-0.924</td>
<td>0.000</td>
<td>0.383</td>
</tr>
</tbody>
</table>

4. Student Research Projects

- **Quantization project:** Recall that once the DCT coefficients are computed, they are rescaled and rounded during the step of quantization. Write a report which explains the quantization of the 2D-DCT coefficients in the JPEG compression algorithm. Given the following DCT coefficients, for an 8 x 8 pixel block of an image, compute the quantized values.
Background: Read the tutorial, “What is Inside a JPEG File”¹² to learn more details about quantization. Note that quantization tables are given for the YCbCr color scheme. The above DCT coefficients were computed for the Y component.

- **Survey project:** Assemble a set of photos and ask classmates to visually assess their authenticity (asking classmates to pay attention to light, shadows, reflections, and perspective).

Background: Read article, “Detecting Fake Photos with Digital Forensics”¹³, an interview with Hany Farid (Mathematician, Computer Science Department at Dartmouth, who works on “statistical tools for detecting tampering in digital images”¹⁴)

  - Interview question: Can average news readers or viewers ever tell whether a photograph has been altered? Are there tell-signs?
  - HF: Yes and no. (Yes, we’re good at detecting inconsistencies like the “floating head syndrome” but, NO, we’re very bad at light and shadows. We basically are unable to tell whether shadows are consistent or inconsistent. “While bad fakes are very easy to detect, good fakes are very difficult to detect; and worse, really good pictures are often said to be fake, because of “this sort of failure to reason about things like lighting and reflections and shadows and perspective.”

- **Fingerprinting Project:** How can we tell whether a particular photo was taken by a particular camera. In other words, how can we tell if we have a match? What is meant by *camera fingerprinting*? What is meant by *sensor fingerprinting*?

Background: Read article, “Binghamton researcher uses photos to link cameras to digital crime,”¹⁵ an article covering Jessica Fridrich (CS, SUNY-Binghamton) who specializes in “data hiding applications in digital imagery, including steganography and steganalysis, forensic analysis of digital images (sensor fingerprints), and advanced image processing.”¹⁶ She notes that this research has applications to determining whether a film is pirated and finding the camera involved in child pornography.

**More background reading:** Read the article, “Detecting Fake Photos with Digital Forensics”¹³ an interview with Hany Farid, a mathematician and digital forensics specialist who teaches computer science at Dartmouth University.

**Hany Farid:** “That’s the beauty of mathematics and physics and computer science; we can quantify and measure whether things are consistent or not. … [But the tools are not at the stage where you can just push a button and get an answer. It’s not like CSI on television. … But there are certain useful techniques. Digital cameras are great for many reasons, but one reason is that they are all really different. … The underlying encoding of a jpeg image varies from camera to camera and we know what the statistics of the jpeg image look like. This technique is useful in a court of law to make sure that digital evidence has not been tampered with and also useful to AP or Reuters when checking whether a photo received from a citizen journalist has been altered.”¹³
Forensics Education and Career Exploration Student Projects

- Develop a web page on Internships in Forensics for undergraduates
  Examples:
  - URI Internships at the Digital Forensics Cyber Security Center
    [http://dfsc.uri.edu/academics/df_internships](http://dfsc.uri.edu/academics/df_internships) (source: Dr. Victor Fay-Wolfe, director, Digital Forensics Cyber Security Center)
  - National Security Agency
    [https://www.nsa.gov/careers/opportunities_4_u/students/](https://www.nsa.gov/careers/opportunities_4_u/students/)
  - Department of Homeland Security
    [http://www.dhs.gov/job-opportunities-students](http://www.dhs.gov/job-opportunities-students)

- Organize and host a Forensics Career Exploration event with guests to include: alumni/alumnae who work in Forensics, staff from the Career Education Center, local forensics experts

- Organize and host a Cyber Security Awareness Week (CSAW) conference
  Example: See the annual CSAW Conference at NYU-Poly held from November 12-15, 2014
References