Using the Singular Value Decomposition Particularly for the Compression of Color Images

Bethany Adams and Nina Manual

November 13, 2005

Abstract

The use of the singular value decomposition for image compression is common; however, its study in the area of true color images is slightly less frequent. In this article, we will delve into the use, history, and definition of the SVD specifically focusing on its use for compression of color images.

1. Introduction

In linear algebra, the idea of the matrix, including its rank, action on vectors, and geometric structure, are integral aspects of the mathematical study area. Because of this, the Singular Value Decomposition (SVD) is very widely used and incredibly important. The SVD can be used in a huge variety of applications including least square approximations to solving systems of linear equations which all have to something to do with the rank of the matrix. The SVD is very good at approximating matrices of specific rank and can, therefore, be quite applicable in these areas. Specifically, throughout this project, we will be studying the use of the SVD's ability to expose the underlying geometry of an image matrix and make a comparable approximation of the parent matrix.

The use of images in software, on web pages, and in many other programs has become so common that we can seldom view anything without their overwhelming presence. From photographs to cliparts, the storage space required for the hefty load of a color image can be overwhelming for a computers' memory and storage facilities. In light of the definitive need for a method of compression that can retain an image's sharpness, color, and original look, many different ways to effectively improve our abilities for storage have been studied. Of these, the singular value decomposition is of

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

particular interest to us. With the use of linear algebra in the factoring of an image matrix A, we can greatly decrease the amount of memory needed to store a "pretty good" approximation of the image in a new image matrix. Throughout this project we will study the use of the SVD and its significance pertaining to the change it makes to an image.

2. History and Development of the SVD

The singular value decomposition was originally developed independently by two mathematicians, Eugenio Beltrami and Camille Jordan, in the mid to late 1800's. The derivation introduced by Beltrami has claim to being the first publication made about the SVD; however, his algorithms were limited in the sense that, in reducing the bilinear form to the linear form, he does not take into consideration degenerate matrices. This problem was remedied by the propositions of Camille Jordan which used a technique known as deflation, a less widely known technique, that remedied many of the difficulties encountered by Beltrami. Several other mathematicians took part in the final developments of the SVD including James Joseph Sylvester, Erhard Schmidt, and Hermann Weyl who studied the SVD into the mid-1900's.

3. Definition of the Singular Value Decomposition

The SVD itself requires some kind of basic knowledge of the terms and processes of linear algebra. With this said, it may be important to begin by assuming that A is an $m \times n$ matrix where $m > n$ into a set of three matrices of the form

$$A_{m\times n} = U_{m\times m} S_{m\times n} V^T_{n\times n} \tag{1}$$

where U and V are orthogonal matrices, meaning that all of the columns of U and V are mutually orthogonal and have length one, and S is a matrix containing the singular values of A in descending order

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$$

along the main diagonal. The matrices U and V are called the singular vectors of A.

When finding the singular values of a matrix, one must first begin by finding the eigenvalues and eigenvectors of two matrices which are derived from A including AA^T and A^TA. The eigenvectors of AA^T are known as the gene coefficient vectors or left singular vectors and make up the columns of the matrix U. However, the eigenvectors of A^TA are called expression level vectors or right singular vectors (as the matrix A is the right multiplier in the expression A^TA) and make up the columns of V. The singular values are found by taking the square root of each of the eigenvalues and placing them down the diagonal of matrix S in descending order as shown previously.

3.1. Finding the eigenvalues and eigenvectors

Assuming that the reader is familiar with basic terminology and functions associated with linear algebra, we can review the methods for obtaining eigenvalue-eigenvector pairs and, finally, the singular values of a matrix.

The eigenvalues of A are all values of λ that solve the equation $A\mathbf{x} = \lambda\mathbf{x}$ where λ is an eigenvalue and $\mathbf{x}$ is its corresponding eigenvector pair. From this equation we can derive the following statement

$$\begin{aligned} A\mathbf{x} &= \lambda\mathbf{x} \\ A\mathbf{x} - \lambda\mathbf{x} &= 0 \\ A\mathbf{x} - \lambda I\mathbf{x} &= 0 \\ (A - \lambda I)\mathbf{x} &= 0. \end{aligned} \tag{2}$$

We can see from this that the eigenvectors are in the nullspace of $A - \lambda I$ meaning that the rows of $A - \lambda I$ are singular and $|A - \lambda I| = 0$. It is in this that the eigenvalues take their form. By subtracting λI from the matrix A and taking the determinant we can produce a polynomial containing λ called the characteristic polynomial. By setting the characteristic polynomial equal to zero we can solve for the λ values which make the $|A - \lambda I| = 0$. Consider the matrix

$$A = \begin{bmatrix} 3 & 6 \\ 1 & -2 \end{bmatrix}. \tag{3}$$

We can find $A - \lambda I$ essentially by subtracting λ from all of the diagonal elements of Matrix A; therefore,

$$A - \lambda I = \begin{bmatrix} 3 - \lambda & 6 \\ 1 & -2 - \lambda \end{bmatrix} \tag{4}$$

and the determinant of $A - \lambda I$ can be computed quite simply.

$$\begin{aligned} |A - \lambda I| &= \begin{vmatrix} 3 - \lambda & 6 \\ 1 & -2 - \lambda \end{vmatrix} \\ &= (3 - \lambda)(-2 - \lambda) - (6)(1) \\ &= \lambda^2 - \lambda - 12 \end{aligned} \tag{5}$$

This is the characteristic polynomial. Using these λ values that are located on this parabola, we can set the determinant equal to any value that we choose. Because we need to compute the nullspace of our matrix, we must force our determinant to be zero and should, therefore, set the characteristic

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

polynomial to zero and solve for our λ's.

$$\begin{aligned}\lambda^2 - \lambda - 12 &= 0\\ (\lambda+3)(\lambda-4) &= 0\\ \lambda_1 &= -3\\ \lambda_2 &= 4\end{aligned} \tag{6}$$

These are our eigenvalues. The computation of the eigenvectors follows naturally from these. Notice that we can now substitute our eigenvalues into the equation $(A-\lambda I)\mathbf{x} = 0$ and solve for the vectors x that are in the nullspace of $(A-\lambda I)$. These $\mathbf{x}$'s will be our corresponding eigenvectors.

$$\begin{aligned}(A-\lambda I)\mathbf{x} &= 0\\ \begin{bmatrix}3-\lambda & 6\\ 1 & -2-\lambda\end{bmatrix}\mathbf{x} &= 0\\ \begin{bmatrix}3-\lambda_1 & 6\\ 1 & -2-\lambda_1\end{bmatrix}\mathbf{x}_1 &= 0\\ \begin{bmatrix}3-(-3) & 6\\ 1 & 1-(-3)\end{bmatrix}\mathbf{x}_1 &= 0\\ \begin{bmatrix}6 & 6\\ 1 & 1\end{bmatrix}\mathbf{x}_1 &= 0\\ \mathbf{x}_1 &= \begin{bmatrix}-1\\ 1\end{bmatrix}\end{aligned} \tag{7}$$

Similarly, we can compute that $\mathbf{x}_2 = \begin{bmatrix}6\\ 1\end{bmatrix}$. It may be helpful to show that $A\mathbf{x} = \lambda\mathbf{x}$. Let's consider only λ_1 for the sake of propriety. Then,

$$\begin{aligned}A\mathbf{x} &= \lambda\mathbf{x}\\ \begin{bmatrix}3 & 6\\ 1 & -2\end{bmatrix}\mathbf{x}_1 &= \lambda_1\mathbf{x}_1\\ \begin{bmatrix}3 & 6\\ 1 & -2\end{bmatrix}\begin{bmatrix}-1\\ 1\end{bmatrix} &= (-3)\begin{bmatrix}-1\\ 1\end{bmatrix}\\ \begin{bmatrix}3\\ -3\end{bmatrix} &= \begin{bmatrix}3\\ -3\end{bmatrix}.\end{aligned} \tag{8}$$

There is, however, a quicker method for deriving the characteristic polynomial if we have a 2×2 matrix. When using this method, we can cut out several steps in the previous computation including

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

the subtraction of λ from all the diagonal elements and the calculation of the determinant of the matrix $A - \lambda I$. The characteristic polynomial can be written as $P(\lambda) = \lambda^2 - T\lambda + D$ where T can be defined as the "trace" of the matrix and D is the determinant of the matrix. With a matrix

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

the trace (T) equals $a + d$ and the determinant equals $ad - bc$.

$$\begin{aligned} T &= a + d \\ D &= ad - bc \end{aligned} \tag{9}$$

Using this we can easily see that the trace of matrix A is $T_A = 3 + (-2) = 1$ and determinant of matrix A is $D_A = (3)(-2) - (6)(1) = -12$. Therefore, the characteristic polynomial of the matrix in equation (3) is $P(\lambda) = \lambda^2 - \lambda - 12$ which, we can see is identical to equation (5). Very powerful.

With an understanding of the computation of an eigenvalue-eigenvector pair, we can more easily understand the steps taken in finding the singular values of a matrix; which, as defined previously, are the square roots of the eigenvalues of the matrices AA^T and A^TA.

3.2. Executing the Singular Value Decomposition

The first step in finding the SVD is to compute AA^T and A^TA. In order to simplify the manual computations needed for completing the SVD, we might find it easier to redefine the matrix as

$$A = \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix}. \tag{10}$$

Therefore, the matrices AA^T and A^TA are

$$\begin{aligned} AA^T &= \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix}^T \\ &= \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ -2 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 8 & 0 \\ 0 & 2 \end{bmatrix} \end{aligned} \tag{11}$$

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

and

$$
\begin{aligned}
A^T A &= \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix}^T \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix} \\
&= \begin{bmatrix} 2 & 1 \\ -2 & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix} \\
&= \begin{bmatrix} 5 & -3 \\ -3 & 5 \end{bmatrix}
\end{aligned}
\tag{12}
$$

and it is here that the SVD begins.

3.2.1. Finding the Matrix U for the Singular Value Decomposition

The eigenvectors of AA^T make up the matrix U.

$$
AA^T = \begin{bmatrix} 8 & 0 \\ 0 & 2 \end{bmatrix}
$$

Using the equation $P(\lambda) = \lambda^2 - T\lambda + D$ we can quickly see that $P(\lambda) = \lambda^2 - 10\lambda + 16$. This polynomial factor quite easily just as the previous examples, so in order to find $0 = \lambda^2 - 10\lambda + 16$ we can simply factor.

$$
\begin{aligned}
0 &= \lambda^2 - 10\lambda + 16 \\
&= (\lambda - 8)(\lambda - 2) \\
\lambda_1 &= 8 \\
\lambda_2 &= 2
\end{aligned}
\tag{13}
$$

Computing the eigenvectors which correspond to these eigenvalues is not very difficult. We must, again, find the nullspace of $A - \lambda I$ which, in our case, is $AA^T - \lambda I$.

$$
\begin{aligned}
(AA^T - \lambda_1 I)\mathbf{x} &= 0 \\
\begin{bmatrix} 8 & 0 \\ 0 & 2 \end{bmatrix} - 8 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x} &= 0 \\
\begin{bmatrix} 8-8 & 0 \\ 0 & 2-8 \end{bmatrix} \mathbf{x} &= 0 \\
\begin{bmatrix} 0 & 0 \\ 0 & -6 \end{bmatrix}
\end{aligned}
\tag{14}
$$

Because we forced the matrix to be singular by setting the determinant equal to zero, we be sure that the nullspace of the matrix is non-trivial. In our case

$$null(AA^T - \lambda_1 I) = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{15}$$

We can do the same sort of computation to find the second eigenvector $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ So, therefore, the eigenvectors are

$$\mathbf{v}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \text{and} \qquad \mathbf{v}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{16}$$

At this point we have, essentially, found the columns of the matrix U but in order to find the final columns, we must have eigenvectors with length one. So we have to unit vectors pointing in the direction of the eigenvectors. This entails division by the vectors magnitude.

$$U_{\mathbf{v}_1} = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} = \frac{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}{\left\|\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\|} = \frac{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}{\sqrt{1}} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{17}$$

$$U_{\mathbf{v}_2} = \frac{\mathbf{v}_2}{\|\mathbf{v}_2\|} = \frac{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}{\left\|\begin{bmatrix} 0 \\ 1 \end{bmatrix}\right\|} = \frac{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}{\sqrt{1}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{18}$$

These are the actual columns of our matrix U.

3.3. Finding the Matrix V

The eigenvectors of A^TA make up the matrix V. As seen in equation (12) our matrix is

$$A^TA = \begin{bmatrix} 5 & -3 \\ -3 & 5 \end{bmatrix}.$$

Using the equation $P(\lambda) = \lambda^2 - T\lambda + D$ we can quickly see that $P(\lambda) = \lambda^2 - 10\lambda + 16$, the same characteristic polynomial as that found for AA^T. In order to find $0 = \lambda^2 - 10\lambda + 16$ we will need to, again, factor the polynomial. We know that the result will be

$$\lambda_1 = 8 \qquad \text{and} \qquad \lambda_2 = 2. \tag{19}$$

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

When computing the eigenvectors that correspond to these eigenvalues we will use a similar approach as before.

$$
\begin{aligned}
(A^TA - \lambda_1 I)\mathbf{x} &= 0 \\
\begin{bmatrix} 5 & -3 \\ -3 & 5 \end{bmatrix} - \lambda_1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x} &= 0 \\
\begin{bmatrix} 5-8 & -3 \\ -3 & 5-8 \end{bmatrix} \mathbf{x} &= 0 \\
\begin{bmatrix} -3 & -3 \\ -3 & -3 \end{bmatrix} \mathbf{x} &= 0
\end{aligned} \tag{20}
$$

It is easy to see that

$$
null(AA^T) = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \tag{21}
$$

We can use the same methods to determine the second eigenvector so, therefore, the eigenvectors are

$$
\mathbf{v}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \qquad \text{and} \qquad \mathbf{v}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \tag{22}
$$

Again, we need unit eigenvectors for the columns of V.

$$
V_{\mathbf{v}_1} = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} = \frac{\begin{bmatrix} 1 \\ -1 \end{bmatrix}}{\left\|\begin{matrix} 1 \\ -1 \end{matrix}\right\|} = \frac{\begin{bmatrix} 1 \\ -1 \end{bmatrix}}{\sqrt{2}} = \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix} \tag{23}
$$

Similarly,

$$
V_{\mathbf{v}_2} = \frac{\mathbf{v}_2}{\|\mathbf{v}_2\|} = \frac{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}{\left\|\begin{matrix} 1 \\ 1 \end{matrix}\right\|} = \frac{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}{\sqrt{2}} = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}. \tag{24}
$$

At this point we have found both the U and V matrices that make up the outer portions of the SVD. The inner matrix S is made up of the singular values of the matrix A. The singular values are the square root of the eigenvalues in descending order.

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

3.4. Finding the Singular Values and the Matrix S

Because both of the matrices AA^T and A^TA produced the same eigenvalues, we will have only two singular values. The matrices will be arranged as follows.

$$
\begin{aligned}
A &= USV^T \\
A &= \begin{bmatrix} U_{\mathbf{v}_1} & U_{\mathbf{v}_2} \end{bmatrix} \begin{bmatrix} \sqrt{\lambda_1} & 0 \\ 0 & \sqrt{\lambda_2} \end{bmatrix} \begin{bmatrix} V_{\mathbf{v}_1} & V_{\mathbf{v}_2} \end{bmatrix}^T \\
A &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{8} & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \\
A &= \begin{bmatrix} 2 & -2 \\ 1 & 1 \end{bmatrix}.
\end{aligned}
\tag{25}
$$

Notice that the singular values are placed in descending order and that their eigenvector pairs are in corresponding multiplying positions for matrices. This is very important.

4. Using Matlab for Singular Value Decomposition

Although it may seem as though completing the SVD on small 2×2 matrices isn't very labor intensive, one could imagine that larger matrices, even 3×3 matrices would be much more difficult. This is where computer-aided graphing devices really shine. In Matlab, there are commands which execute the SVD for us. Consider the 4×4 matrix

$$
A = \begin{bmatrix} 1 & 0 & 2 & 1 \\ 0 & 0 & 1 & 1 \\ 3 & 3 & 2 & 1 \\ 1 & 2 & 0 & 3 \end{bmatrix} \tag{26}
$$

By hand, it would take a great deal of computation to first find AA^T andA^TA then to find each eigenvalue for each matrix, then to find all of the corresponding eigenvectors, to divide each eigenvector by its length, and finally to arrange all the vectors in the correct order with singular values in descending positions down the diagonal of the center matrix. Possible, but exhausting. With Matlab we can simply enter the matrix, and type:

```
A=[1 0 2 1;0 0 1 1;3 3 2 1;1 2 0 3]


A =


      1            0            2            1
```


Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

```
      0          0          1          1
      3          3          2          1
      1          2          0          3

EDU>> [U,S,V]=svd(A)

U =

  -832/2731     272/797      18/25      227/435
  -139/944    -158/4191   1711/3108   -527/642
  -345/443     391/894    -793/1890   -171/1057
  -440/833    -955/1149    257/5152    381/2290


S =

  5672/961        0          0          0
      0       875/352        0          0
      0          0      1123/568        0
      0          0          0       401/1454


V =

  -603/1123    557/1684    -45/182     1177/1600
  -941/1637   -359/2548  -1163/1984    -670/1211
  -287/732    1049/1716    442/759     -415/1136
  -819/1717   -913/1295    790/1561     495/3563
```

Matlab returns a U, S, and V matrix directly. Because of the ease of the SVD computation in Matlab, we have a wonderful opportunity to complete decomposition of much larger image matrices.

5. The Singular Value Decomposition's Effect on an Image's Corresponding Matrix

Images on computers are obviously represented by a set of ordered numbers arranged in a matrix. Each number in the matrix represents a pixel in the image or maps to a "color map" which will chart the corresponding color of the pixel in the image. When we decompose an image matrix, we have the ability to isolate certain portions of the image because of the arrangement of the singular values. The idea of the image compression comes from the fact that, in a real image matrix, we would commonly have hundreds of singular values, some of which are so small that they are negligible and can be discarded without losing much clarity. Consider the following proof:

$$\begin{aligned}
A &= USV^T \\
&= \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 & \cdots & \mathbf{u}_n \end{bmatrix} \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \mathbf{v}_3^T \\ \vdots \\ \mathbf{v}_n^T \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{u}_1\sigma_1 & \mathbf{u}_2\sigma_2 & \mathbf{u}_3\sigma_3 & \cdots & \mathbf{u}_n\sigma_n \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \mathbf{v}_3^T \\ \vdots \\ \mathbf{v}_n^T \end{bmatrix} \\
&= \mathbf{u}_1\sigma_1\mathbf{v}_1^T + \mathbf{u}_2\sigma_2\mathbf{v}_2^T + \mathbf{u}_3\sigma_3\mathbf{v}_3^T + \cdots + \mathbf{u}_n\sigma_n\mathbf{v}_n^T
\end{aligned} \tag{27}$$

Each of the terms in this equation represent matrices that, when added together, will equal the original matrix. The interesting thing to note is that the σ terms that are located toward the end of the summation are representing the singular values that are furthest down the matrix meaning that they are the smallest values. In fact, some of the singular values are so small that the terms in which they appear are virtually forced to zero and can be discarded. It is here that we can eliminate some of the storage required for our image matrix without losing important information.

Using Matlab we can write certain codes which execute the SVD on an image matrix and which allow us to view the decomposed image containing only the first few or first several singular values. We can compare the clarity of the decomposed image to the original after each iteration, meaning after each additional singular value is added, and can decide at which point the decomposed image is similar enough to act as a sufficient representation. Often, we can eliminate hundreds of singular

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

values and still produce an almost identical image. This is what is so amazing about the SVD; we can maintain a constant size and clarity of an image while cutting a substantial amount of the necessary information from within the matrix, reducing the storage space needed drastically.

6. SVD of a Color Image

Color images are slightly more complicated to decompose because a true color image is comprised of three layers of matrices that represent respectively the amount of red, green, and blue in each pixel. Therefore, every pixel has three corresponding numbers which make up its actual color. The SVD does not handle multidimensional matrices so in order to decompose a color image, another method must be developed. Matlab allows us to isolate each layer of the image matrix and decompose it separately so it is a very effective tool when dealing with color (or any) image decompression. We use the following Matlab code for decompression of an image:

```
clear Q
close all
L=imread('image.jpg');
imshow(L)
L1=L(:,:,1);
L2=L(:,:,2);
L3=L(:,:,3);

I1=im2double(L1);
I2=im2double(L2);
I3=im2double(L3);

figure imshow(I1)
figure imshow(I2)
figure imshow(I3)

[u1,s1,v1]=svd(I1);
[u2,s2,v2]=svd(I2);
[u3,s3,v3]=svd(I3);
C1=zeros(size(I1));
C2=zeros(size(I2));
C3=zeros(size(I3));
k=16;
for j=1:k
```

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

```
   C1=C1+s1(j,j)*u1(:,j)*v1(:,j).';
end
for j=1:k
   C2=C2+s2(j,j)*u2(:,j)*v2(:,j).';
end
for j=1:k
   C3=C3+s3(j,j)*u3(:,j)*v3(:,j).';
end

figure imshow(C1)
figure imshow(C2)
figure imshow(C3)

C1(k)=1;
C2(k)=1;
C3(k)=1;
R1=im2uint8(C1);
R2=im2uint8(C2);
R3=im2uint8(C3);

Q(:,:,1)=R1;
Q(:,:,2)=R2;
Q(:,:,3)=R3;
figure
imshow(Q,[])
```

Notice that the beginning code peels the red, green, and blue layers from the original image. These layers are imaged separately in the file so that we can study their individual influence on the original image. We must then convert the matrices to class double because, in Matlab this is the only class of image matrix that we can SVD. Finally, we are able to individually decompose each layer and recompile a final, 8-bit, true color, decomposed image. The k-value in the m-file represents the number of iterations taken on each layer, or the number of singular values used in the resulting decomposition. This is actually the rank of the SVDed matrix. By increasing the rank we can increase clarity until an optimal image is reached.

Consider the test image Figure 1, the photograph is a true color image which contains many of the common attributes of a regular photo, buildings, sudden and gradual color change, humans, and realistic landscape or background. It would be easy to identify any discrepancies between our original image and our decomposed image because of the photograph's intricacies and this makes it

Figure 1: Original Image.

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

Home Page

Title Page

◀◀ ▶▶

◀ ▶

Page 14 of 20

Go Back

Full Screen

Close

Quit

(a) Original Image (b) Red Layer

Figure 2: Compare the Colors of the Image to the Red Layer Intensities.

quite desirable as a test image. If we run the file through our previously printed Matlab m-file, we can see that the three layers in double (black and white) form can be viewed next to their parent photo. Each "doubled" layer, Figure 2 and Figure 3, shows a different intensity for the same pixels of the original image because each layer only shows the amount of one color in the pixel.

By viewing the progression of the final recompiled image after an increased iteration, we can see at which point the rank of the decomposed matrix is high enough to effectively imitate the original matrix. It may be helpful to see some iterations over our test image. Consider Images 4 through 5.

The rank of our original matrix is 320 but the rank of this final compressed version is only 100. It is quite amazing the amount of information that can be cut from the matrix while still maintaining such a close imitation of the original photograph. Notice the striking similarity between the original image and the final image iteration with rank 100 shown in Figure 6. We are able to store the image with only a percentage of the starting information, in fact, only 31.25 percent of the original information. This is quite handy when we need to store ten or twenty images on one web page, or we need more room for text, or quicker download time. The SVD is an effective and amazing tool for compression of storage requirements.

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

Home Page
Title Page
◀◀ ▶▶
◀ ▶
Page 15 of 20
Go Back
Full Screen
Close
Quit

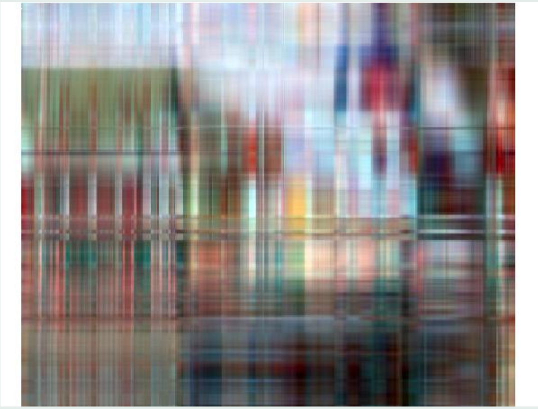
(a) Green Layer (b) Blue Layer

Figure 3: Compare the Colors of the Image to the Green and Blue Layer Intensities.

(a) k=1 (b) k=2

Figure 4: Early Iterations.

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion



(a) k=5

(b) k=7

(a) k=10

(b) k=12

Introduction
History and ...
Definition of the ...
Using Matlab for ...
The Singular Value ...
SVD of a Color Image
Conclusion

(a) k=15

(b) k=20

(a) k=30

(b) k=50

(a) k=75 (b) k=100

Figure 5: Final Iterations.

(a) Original Image (k=320) (b) Final Iteration (k=100)

Figure 6: Compare the Decomposed Image to the Original.

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion

7. Conclusion

Although, the singular value decomposition is more or less a basic factorization of some matrix into its component pieces, it opens a door to us in the world of image compression. The SVD allows us to arrange the portions of a matrix in, essentially, order of importance. It follows that the most important singular values will produce the most important unit eigenvectors. By truncating our matrix summation at the point at which the singular values become too small to greatly effect the final image, we can eliminate large portions of our matrix without losing purpose or clarity. Much of this stems from the fact that the human eye is incapable of detecting minuscule changes in colors which are close to one another. We are then able to simply eliminate the unnecessary intricacies of the image and the SVD makes it possible to isolate these intricacies and remove them.

References

[1] Hourigan, Jody S., and Lynn V. McIndoo. "The Singular Value Decomposition." Dec. 1998. College of the Redwoods. 16 Dec. 2005 http://online.redwoods.cc.ca.us/instruct/darnold/LAPROJ/Fall98/JodLynn/report2.pdf.

[2] Stewart, G. W. "On the Early History of the Singular Value Decomposition." Dec. 1993. College of the Redwoods. 16 Dec. 2005 http://www.ifi.uio.no/~inf9540/SVD.pdf.

[3] Arnold, Ben. "An Investigation into using Singular Value Decomposition as a Method of Image Compression." Sept. 2000. Dept. of Mathematics and Statistics, Universtiy of Canterbury. 16 Dec. 2005 http://www.math.ucalgary.ca/~laf/teaching/Material/Arnold.pdf.

[4] Strange, Gilbert. *Introduction to Linear Algebra.* Wellesley-Cambridge Press.

[5] Dave Arnold.

Introduction
History and . . .
Definition of the . . .
Using Matlab for . . .
The Singular Value . . .
SVD of a Color Image
Conclusion