جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

كلية العلـــوم
قـســـــم الانـــظـــمـــة الـــطـبـيـة الـــذكـــيـــة

المحاضرة السابعة

Matrix Decomposition Methods in Image Processing

المادة : DSP
المرحلة : الثالثة
اسم الاستاذ: م.م. ريام ثائر احمد

1. **Some Types of Matrix Decomposition Methods**
2. **The Effect of the Matrix Decomposition Methods on Images**
    a. **The Effect of SVD Method on image**
    b. **The Effect of Hessenberg Decomposition Method on image**
    c. **The Effect of QR Decomposition Method on image**
    d. **The Effect of LU Decomposition Method on image**
3. **Singular Value Decomposition (SVD) and Image Processing**
4. **Singular Value Decomposition Method Mathematically**
5. **Computing the SVD by Hand**
6. **Theorem**
7. **Some Singular Value Decomposition (SVD) Properties in DIP**

Matrix Decomposition Methods in Image Processing

In the linear algebra discipline in mathematical, a matrix decomposition or matrix factorization is a factorization of a matrix into a product of matrices. There are many different matrix decompositions; each finds use among a particular class of problems.

- ## Some Types of Matrix Decomposition Methods

a. Decompositions based on eigenvalues and related concepts

a.1 Singular value decomposition

a.2 Hessenberg decomposition

a.3 Eigen decomposition

a.4 Jordan decomposition

a.5 Schur decomposition

a.6 Real Schur decomposition

a.7 QZ decomposition

a.8 Takagi's factorization

a.9 Scale-invariant decompositions

b. Decompositions related to solving systems of linear equations

b.1 QR decomposition

b.2 LU decomposition

b.3 Cholesky decomposition

b.4 LU reduction

b.5 Block LU decomposition

Methods for matrix decomposition in linear algebra have found numerous applications in image processing. Therefore, it seems reasonable to investigate matrix decomposition applications in image processing. The following is a list of more applications of matrix decomposition methods:

- Data Compression
- CT Scan Reconstruction
- Linear Regression (Least Square Systems)
- Spectral Clustering
- Moore-Penrose Pseudo Inverse
- Signal Estimation Theory
- Derivation of the Recursive Least-Squares Filter
- Sensor Array Signal Processing

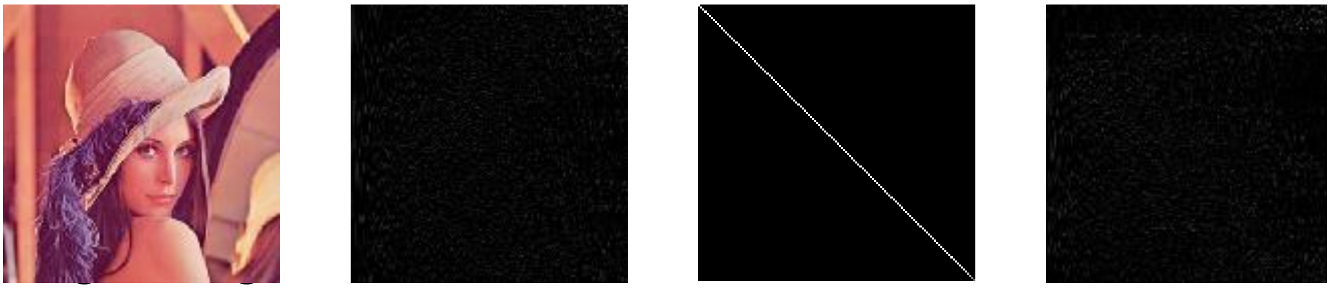
In addition, in machine learning and statistics, we often have to deal with structural data, which is generally represented as a table of rows and columns, or a matrix. A lot of problems in machine learning can be solved using matrix algebra and vector calculus. Applications covered are background Removal, topic modeling, recommendations using collaborative filtering and eigenfaces.

2. The Effect of the Matrix Decomposition Methods on Images:

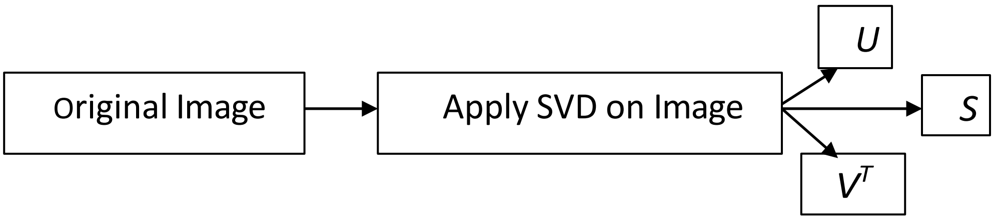
A multitude of matrix decomposition techniques stemming from linear algebra to have been applied to image processing. When the decomposition methods are performed on the whole image, the result will be as the following.

a- The Effect of SVD Method on image

$$I=USV^T$$

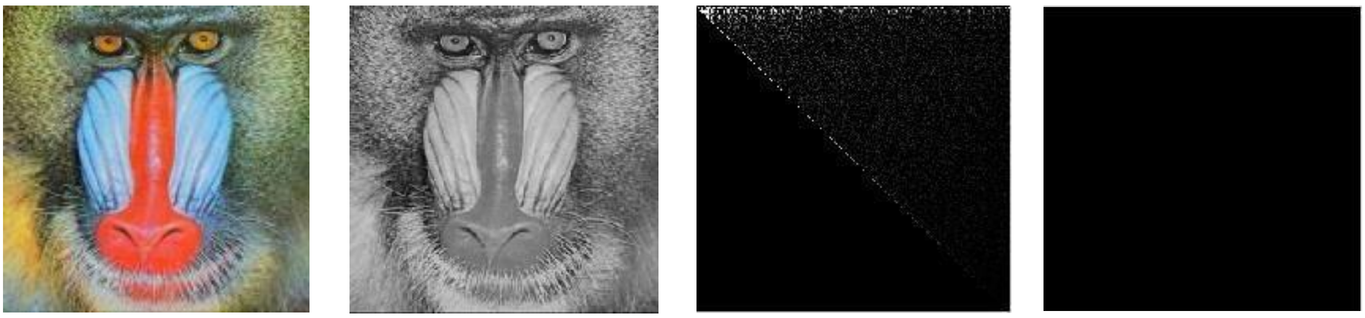


According to the following algorithm:



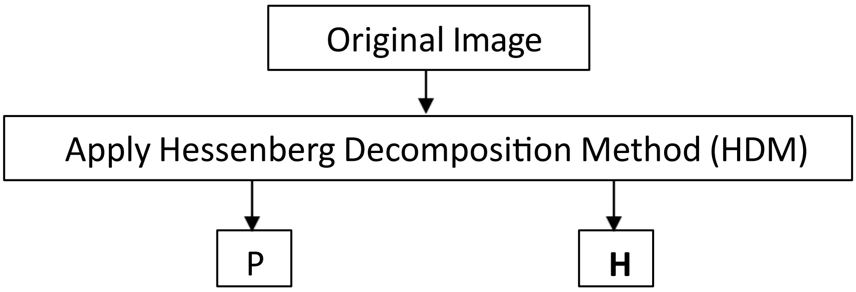
b- <u>The Effect of Hessenberg Decomposition Method on image</u>

$$A = P\,H\,P^{T}$$



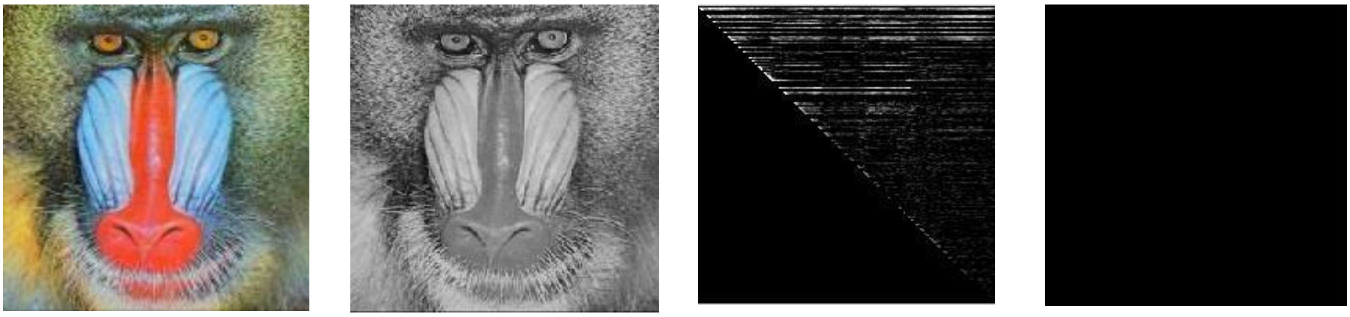
Original Image **Gray** **H-matrix** **P-matrix**

According to the following figure:



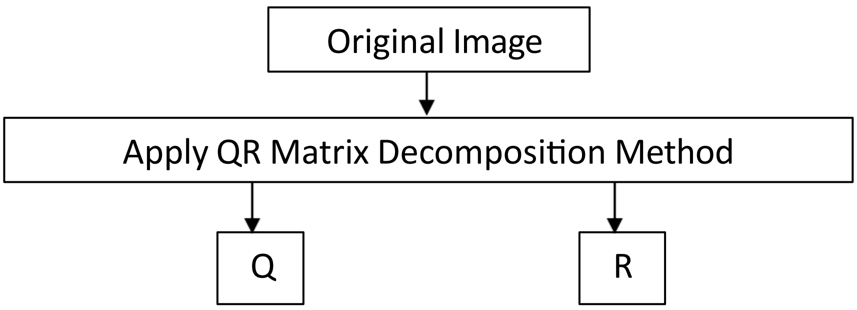
c- The Effect of QR Decomposition Method on image

$$A = Q\,R$$



Original Image Gray R-matrix Q-matrix

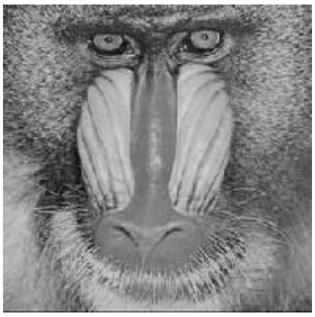
According to the following figure:



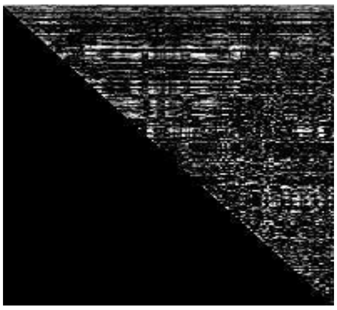
d- The Effect of LU Decomposition Method on image

$$A = L\,U$$

Original Image



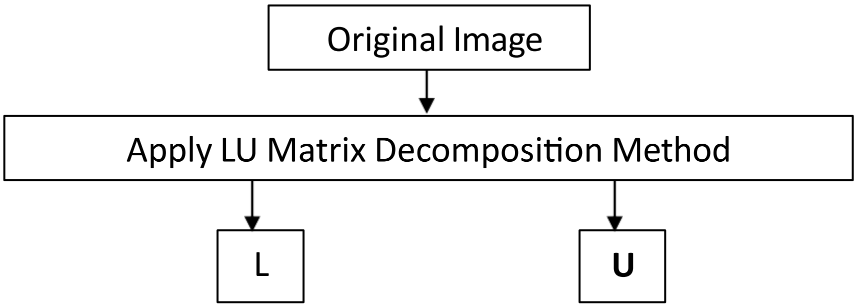
Gray



U-matrix

L-matrix

According to the following algorithm:



3. The Singular Value Decomposition (SVD) and Image Processing

Singular Value Decomposition (SVD) has recently emerged as a new paradigm for processing different types of images. SVD is an attractive algebraic transform for image processing applications.

In linear algebra, the SVD is a factorization of a rectangular real or complex matrix analogous to the diagonalizations of symmetric or Hermitian square matrices using a basis of eigenvectors. SVD is a stable and effective method to split the system into a set of linearly independent components, each of them bearing own energy contribution.

In digital image processing, image features are divided into four groups: visual features, statistical pixel features, transform coefficient features, and algebraic features. The SVD technique can be considered as an algebraic feature. The algebraic, usually represent intrinsic properties.

SVD method can transform matrix A into product USV^T, which allows us to refactor a digital image in three matrices. The use of singular values of such refactoring allows us to represent the image with a smaller set of values, which can preserve useful features of the original image, but use less storage space in the memory, and achieve the image compression process.

The objective of this section is to apply linear algebra "Singular Value Decomposition (SVD)" to mid-level image processing, such as image compression and recognition. The method is factoring a matrix A into three new matrices U,S, and V, in such a way that A=USV^T. Where U and V are orthogonal matrices and S is a diagonal matrix.

4. Singular Value Decomposition Method Mathematically:

Let A be any $m \times n$ matrix. Then there are orthogonal matrices U, and a diagonal matrix S such that

$$A = USV^T = \vec{u}_1\sigma_1\vec{v}_1^T + \vec{u}_2\sigma_2\vec{v}_2^T + \cdots + \vec{u}_r\sigma_r\vec{v}_r^T = \sum_{i=1}^{r} \vec{u}_i\sigma_i\vec{v}_i^T .$$

5. Computing the SVD by Hand:

We now list a simplistic algorithm for computing the SVD of a matrix A. It can be used fairly easily for manual computation of small examples. For a given $m \times n$ matrix A the procedure is as follows:

Example: The SVD form of the matrix *is:*

-

$$A=\begin{pmatrix}5 & 5\\-1 & 7\end{pmatrix}$$

1- Find A^TA.

$$A^TA=\begin{pmatrix}5 & -1\\5 & 7\end{pmatrix}\begin{pmatrix}5 & 5\\-1 & 7\end{pmatrix}=\begin{pmatrix}26 & 18\\18 & 74\end{pmatrix}$$

2- To find the eigenvalue,

$$\det(A^TA-\lambda I)=\left|\begin{pmatrix}26 & 18\\18 & 74\end{pmatrix}-\begin{pmatrix}\lambda & 0\\0 & \lambda\end{pmatrix}\right|=\lambda^2-100\lambda+1600$$

$$(\lambda-80)(\lambda-20)=0,\ \lambda_1=80,\ \lambda_2=20$$

3- The singular value equals $\sigma_i=\sqrt{\lambda_i}$

$$\sigma_1=\sqrt{80},\ \sigma_2=\sqrt{20}$$

4- To find the eigenvector v_1 and v_2

To find the eigenvector v_1:

$$(A^TA-\lambda_1 I)\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$(A^TA\text{- }80I)\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$\left(\begin{pmatrix}26 & 18\\18 & 74\end{pmatrix}-\begin{pmatrix}80 & 0\\0 & 80\end{pmatrix}\right)\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$\begin{pmatrix}-54 & 18\\18 & -6\end{pmatrix}\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$-54x+18y = 0$$

$$y = 54x/18$$

If $x=1$, then $y=3$.

$$X_1=\begin{pmatrix}1\\3\end{pmatrix}$$

$$|X_1|=\sqrt{1^2+3^2}=\sqrt{10}$$

$$v_1=X_1\,/\,|X_1|$$

$$v_1=\begin{pmatrix}1/\sqrt{10}\\3/\sqrt{10}\end{pmatrix}$$

To find the eigenvector v_2:

$$(A^TA\text{- }\lambda_2 I)\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$(A^TA\text{- }20I)\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$\left(\begin{pmatrix}26 & 18\\18 & 74\end{pmatrix}-\begin{pmatrix}20 & 0\\0 & 20\end{pmatrix}\right)\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$\begin{pmatrix}6 & 18\\18 & 54\end{pmatrix}\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}$$

$$6x+18y = 0$$

$$x = -18y/6$$

If $y = 1$, then $x = -3$.

$$X_2 = \begin{pmatrix} -3 \\ 1 \end{pmatrix}$$

$$|X_2| = \sqrt{(-3)^2 + 1^2} = \sqrt{10}$$

$$v_2 = X_2 / |X_2|$$

$$v_2 = \begin{pmatrix} -3/\sqrt{10} \\ 1/\sqrt{10} \end{pmatrix}$$

$$V = \begin{pmatrix} 1/\sqrt{10} & -3/\sqrt{10} \\ 3/\sqrt{10} & 1/\sqrt{10} \end{pmatrix}$$

$$V^T = \begin{pmatrix} 1/\sqrt{10} & 3/\sqrt{10} \\ -3/\sqrt{10} & 1/\sqrt{10} \end{pmatrix}$$

5- $S = \begin{pmatrix} 4\sqrt{5} & 0 \\ 0 & 2\sqrt{5} \end{pmatrix}$

6- To find U:

$$u_1 = 1/\sigma_1 A v_1 = 1/4\sqrt{5} \times \begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix} \times \begin{pmatrix} 1/\sqrt{10} \\ 3/\sqrt{10} \end{pmatrix} = 1/4\sqrt{5} \begin{pmatrix} 20/\sqrt{10} \\ 20/\sqrt{10} \end{pmatrix}$$

$$u_2 = 1/\sigma_2 A v_2 = 1/2\sqrt{5} \times \begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix} \times \begin{pmatrix} -3/\sqrt{10} \\ 1/\sqrt{10} \end{pmatrix} = 1/2\sqrt{5} \begin{pmatrix} -10/\sqrt{10} \\ 10/\sqrt{10} \end{pmatrix}$$

$$U = \begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}$$

7- The SVD form of the matrix A is:

$$SVD = U\,S\,V^T = \begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix} \times \begin{pmatrix} 4\sqrt{5} & 0 \\ 0 & 2\sqrt{5} \end{pmatrix} \times \begin{pmatrix} 1/\sqrt{10} & 3/\sqrt{10} \\ -3/\sqrt{10} & 1/\sqrt{10} \end{pmatrix} =$$

$$\begin{pmatrix} 4\sqrt{5}/\sqrt{2} & -2\sqrt{5}/\sqrt{2} \\ 4\sqrt{5}/\sqrt{2} & 2\sqrt{5}/\sqrt{2} \end{pmatrix} \times \begin{pmatrix} 1/\sqrt{10} & 3/\sqrt{10} \\ -3/\sqrt{10} & 1/\sqrt{10} \end{pmatrix} =$$

$$\begin{pmatrix} 2\sqrt{10} & -\sqrt{10} \\ 2/\sqrt{10} & \sqrt{10} \end{pmatrix} \times \begin{pmatrix} 1/\sqrt{10} & 3/\sqrt{10} \\ -3/\sqrt{10} & 1/\sqrt{10} \end{pmatrix} = \begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix} = A$$

2. Theorem:

Any $m \times n$ real matrix A can be factored uniquely into a product of the form USV^T, called the SVD of A, where U and V are orthogonal matrices and S is an $m \times n$ diagonal matrix whose diagonal entries called the singular values of A are all real and satisfy the following:

$$k = \min(m,) \qquad \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k \geq 0$$

Let σ_j denote the j^{th} singular value along the diagonal of S for j=1,...,. If u_j and v_j represent the j^{th} column vectors of U and V, respectively, then A can be written as

$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + _k \qquad \text{(Complete Form) (1)}$$

We can approximate A by matrices of lower rank by truncating the expansion (1). Most of the information contained in A will be reproduced using relatively few terms of the expansion (1). We expect a matrix of the form

$$A_r = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r \qquad \text{(Truncated Form)}$$

to adequately represent the original image given by A even if r is much smaller than k (where r is the number of the largest singular values of A) because we are using the largest singular values first. If $\sigma_r > 0$, then A_r is a rank approximation to A. Students can reconstruct the images using the SVD with different ranks. The total storage for A_r will be

$$T_s(A_r) = r$$

The integer r can be chosen confidently less then n, and the digital image corresponding to A_r still have very close the original image. However, the chose the different r will have a different corresponding image and storage for it. For typical choices of the k, the storage required for A_r will be less the percent.

Using the command *subplot*, they can plot all these approximations along with the original image in the same window for easy comparison.

Moreover, they can compute the error between the original image and its approximations. One way of doing this is through the Frobenius norm of a matrix which is defined as

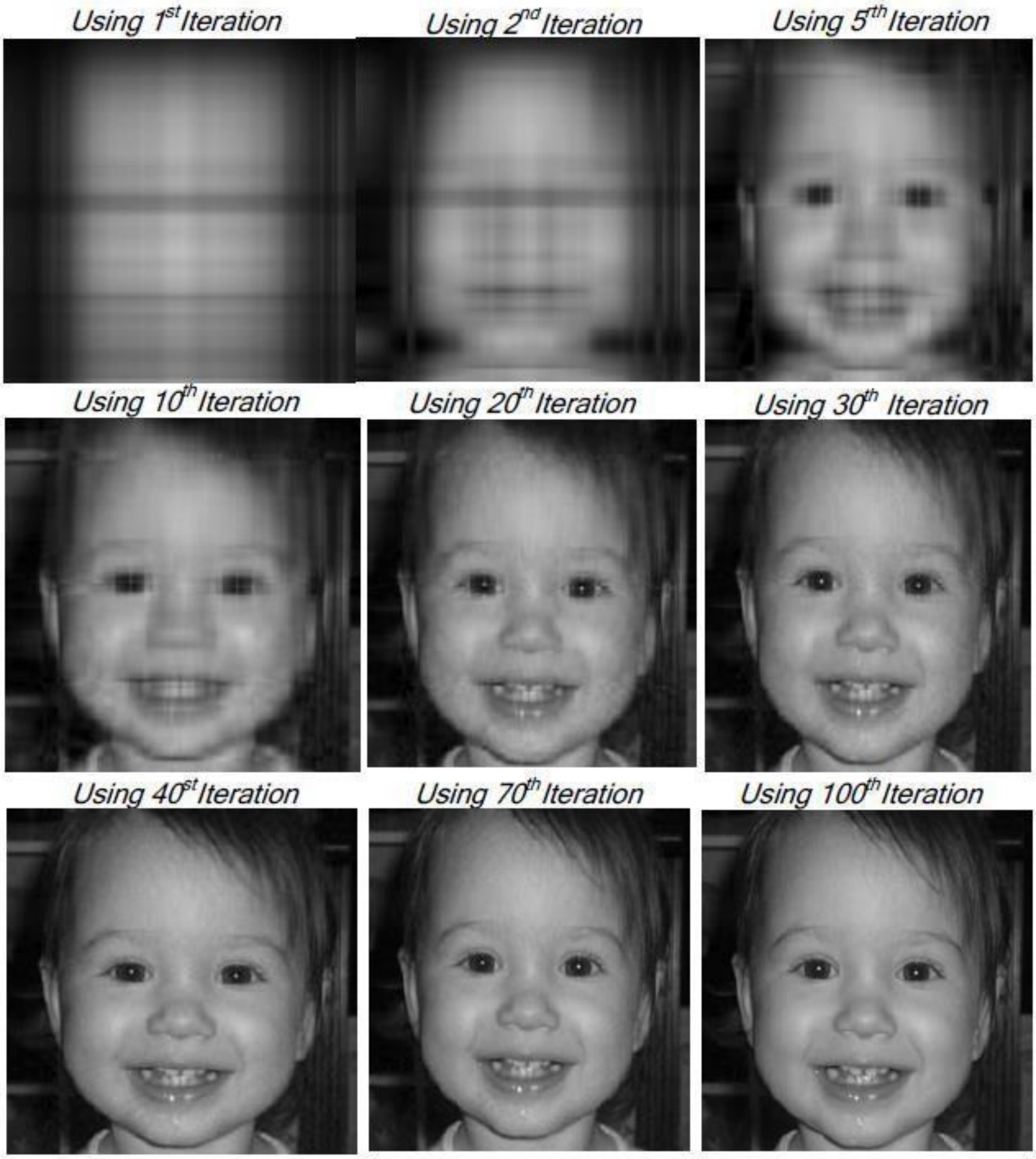
$$||A||_{\text{Frob}} = \sqrt{\sum_{i=}^{m} \sum_{j=}^{n} [A(i,j)]^2}$$

Let A_c represent a compressed version of the image A: We define the relative error as

$$E_{rel} = \frac{||A - A_c||_{\text{Frob}}}{||A||_{\text{Frob}}}.$$

Students can compute the relative error in the Frobenius norm of the image A at different ranks and check if the results of the norm roughly agree with the error based on visual perception. They can investigate, for example, how large does the rank needs to be so that the relative error (in the Frobenius norm) is less than 5%.

Grayscale Original Image of Size 497x498



As we see the 10^{th} Iteration the image contains the 100 entries, also form the 30th Iteration we get the image near to original image and form the 70^{th} Iteration i.e. A 70×70 matrix, with 4900 entries is significantly reduced the original image of size 497×498 matrix, with 247506 entries. So, there is no need to go up to 100^{th} Iteration.

In the following examples, we will show how the SVD works in several applications in DIP.

3. Some Singular Value Decomposition (SVD) Properties in DIP:

The first property of SVD is that:
a- The singular values $\sigma_{1,2,\ldots,}\sigma_n$ are unique, but the matrices U and V are not unique.
b- The SVD method is a robust and reliable orthogonal matrix decomposition method.

Due to SVD conceptual and stability reasons, it becomes more and more popular in the signal processing area. SVD is an attractive algebraic transform for image processing. SVD has prominent properties in imaging. Although some SVD properties are fully utilized in image processing, others still need more investigation and contributed to it.

c- The SVD packs the maximum signal energy into as few coefficients. It has the ability to adapt to the variations in local statistics of an image. However, SVD is an image adaptive transform; the transform itself needs to be represented in order to recover the data.

d- The SVD method decomposes a matrix into orthogonal components with which optimal sub rank approximations may be obtained. The largest object components in an image found using the SVD generally correspond to eigenimages associated with the largest singular values, while image noise corresponds to eigenimages associated with the smallest singular values. The

SVD is used to approximate the matrix decomposing the data into an optimal estimate of the signal and the noise components. This property is one of the most important properties of the SVD decomposition in noise filtering, compression and forensic which could also be treated as adding noise in a proper detectable way.