Handbook of Research on Threat Detection and Countermeasures in Network Security

Alaa Hussein Al-Hamami
Amman Arab University, Jordan

Ghossoon M. Waleed Al-Saadoon
Applied Science University, Bahrain

A volume in the Advances in Information Security, Privacy, and Ethics (AISPE) Book Series

Information Science REFERENCE
An Imprint of IGI Global

Managing Director: Lindsay Johnston
Managing Editor: Austin DeMarco
Director of Intellectual Property & Contracts: Jan Travers
Acquisitions Editor: Kayla Wolfe
Production Editor: Christina Henning
Development Editor: Erin O'Dea
Typesetter: Amanda Smith
Cover Design: Jason Mull

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA, USA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Library of Congress Cataloging-in-Publication Data

Al-Hamami, Alaa Hussein, 1948-
Handbook of research on threat detection and countermeasures in network security / Alaa Hussein Al-Hamami and Ghossoon M. Waleed Al-Saadoon, editors.
pages cm
Includes bibliographical references and index.
ISBN 978-1-4666-6583-5 (hardcover) -- ISBN 978-1-4666-6584-2 (ebook) -- ISBN 978-1-4666-6586-6 (print & perpetual access) 1. Computer networks-Security measures. I. Al-Saadoon, Ghossoon M. Waleed, 1969- II. Title.
TK5105.59.A393 2015
005.8--dc23
2014029335

This book is published in the IGI Global book series Advances in Information Security, Privacy, and Ethics (AISPE) (ISSN: 1948-9730; eISSN: 1948-9749)

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 12
Data Hiding Schemes Based on Singular Value Decomposition

Nidhal Khdhair El Abbadi
University of Kufa, Iraq

ABSTRACT

The security of information exchange is very important on the network. Authentication and information hiding have also become important issues. Information hiding techniques are acquiring an increasing importance due to the widespread diffusion of multimedia contents. The aim of this chapter is to focus on the Singular Value Decomposition (SVD) transform, with the aim of providing an exhaustive overview on those steganography, image cryptography and watermarking techniques leveraging on the important properties of such a transform. Despite the attention it has received in the last years, SVD in image processing and security is still in its infancy. Many SVD characteristics are still unutilized in image processing. In this chapter the author tries to highlight the basic properties of SVD and some of their applications in the field of security to encourage researchers to discover more about SVD properties which are not yet utilized.

INTRODUCTION

Due to the rising dependence on digital media and the unexpected expansion of the distribution opportunities over the Internet, techniques for hiding information into digital contents are achieving significant importance. Such techniques aim to provide the ability to communicate secretly and the capacity to protect copyrighted multimedia content against illegal distribution. Designing such schemes has become a topic of great importance and many researchers have spent much effort in the last years to obtain an effective solution. However, despite many different approaches have been attempted, there is currently no scheme that can preserve imperceptibility of the hidden data while ensuring a high security against malicious attacks.

In networked environments, the safety of multimedia data can be investigated according to two aspects: the safety of static data and the data security during dynamic communication. The safety of static multimedia data can be inspected according to the following four aspects:

DOI: 10.4018/978-1-4666-6583-5.ch012

1. **Storage:** Is the data centrally stored, or dispersed?
2. **Vulnerability:** How robustness is the data against theft or abuse?
3. **Confidence/Authenticity:** What constitutes authentic information? Can that information be tampered with?
4. **Linking:** Will the multimedia data be linked to other information, e.g., about originating and/or consuming party?

When inspecting the security of real-time multimedia communication, one should take into account the specific properties of both multimedia data and real-time communication. First, limited distortions in multimedia data cannot be perceived by end users. Thus some bit errors and packets loss that may occur during communication do not defect the overall visual/audio quality. Secondly, due to scheduling protocols of real-time multimedia communication, packet loss may happen. Thirdly, caused by the large amount of multimedia data, communication security trade-offs should be low enough.

The upcoming information processing architectures for ubiquitous computing is highly sensitive to security issues. For some networked scenarios, such as fingerprint collection in distributed environment, video monitoring and health care systems, the image integrity and authenticity is fatal to the success of these services. While most of the embedded systems working in such distributed environments are low-end devices in terms of their computing power, memory size and communication bandwidths. Therefore, new security policies have to provide, such that the given constraints of these devices are considered accordingly.

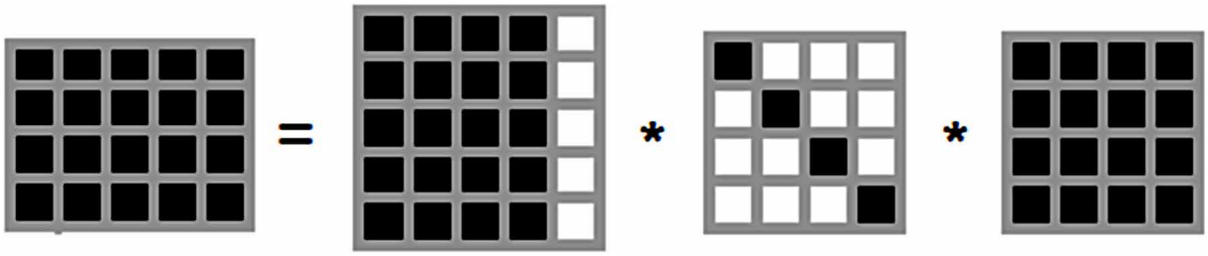
For the current digital age, digital forensic research becomes imperative. Counterfeiting and falsifying digital data or digital evidence with the goal of making illegal profits or bypassing laws is the main objective for the attackers. The forensic research focuses in many tracks; steganography, watermarking, authentication, labeling, captioning, etc. Many applications were developed to satisfy consumer requirements such as labeling, fingerprinting, authentication, copy control for DVD, hardware/ software watermarking, executables watermarks, and signaling (signal information for automatic counting) for the purpose of broadcast monitoring count (Sadek, 2012).

The SVD packs the maximum signal energy into 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. Despite the attention it has received in the last years, SVD in image processing is still in its infancy. Many SVD characteristics are still unutilized in image processing. The present chapter highlights the basic properties of SVD and some of their applications in the field of security to encourage researchers to discover more about SVD properties which are not yet utilized (Liu & Tan, 2002).

Following some objectives of writing this chapter are:

1. Providing readers with knowledge about the SVD, and the usefulness of using SVD in image processing and specifically in security fields.
2. Prompting the researcher to discover new algorithms of using SVD in the fields of security and countermeasures such as steganography, watermarking, image encryption, stegoanalysis ... etc.
3. There are many SVD features not utilized. The SVD chapter may help to discover some of these features and utilize them in the fields of communication security in general.
4. Highlighting some novel uses of SVD in the fields of image and text cryptography (to the best of knowledge, there are no published papers in the field of using SVD in text encryption), steganography, the watermarks, stegoanalysis (which is also new field in using SVD), and some other new applications.

Figure 1. Decomposition of matrix by SVD



Using SVD in the fields of steganography, watermark, image encryption, and stegoanalysis is a new subject. There are few books discussed this issue, and I think this subject is very useful for postgraduate students, professors, researcher, and every one interested in the field of communication security.

SINGULAR VALUE DECOMPOSITION (SVD)

Singular value decomposition for square matrices was discovered by Beltrami in 1873 and Jordan in 1874, and extended to rectangular matrices by Eckart and Young in the 1930s. It was not used as a computational tool until the 1960s because of the need for sophisticated numerical techniques. In later years, Gene Golub demonstrated its usefulness and feasibility as a tool in a variety of applications. SVD is one of the most useful tools of the linear algebra with several applications in image compression, watermarking, and other signal processing areas. In the linear algebra, the singular valued composition (SVD) is a factorization of a real or complex matrix with many useful applications in signal processing and statistics. SVD is to be a significant topic in the linear algebra by many renowned mathematicians. It has many practical and theoretical values. Special features of SVD is that it can be performed on any real (m, n) matrix.

In the linear algebra the SVD is a factorization of a rectangular real or complex matrix analogous to the diagonalization of symmetric or Hermitian square matrices by using a basis of eigenvectors. SVD is a stable and an effective method to split the system into a set of linearly independent components, each of them bearing own energy contribution (Andrews & Patterson, 1976).

DEFINITION OF SVD

For any given matrix $A \in R^{m \times n}$ there exists decomposition $\mathbf{A} = \mathbf{USV^T}$ such that

- U is an $m \times n$ matrix with orthonormal columns.
- S is an $n \times n$ diagonal matrix with non-negative entries.
- V^T is an $n \times n$ orthonormal matrix.

We can visualize this decomposition as in Figure 1.

The diagonal values of S are called 'Singular Values' of **A**.

The column vectors of U are the 'Left Singular Vectors' of **A**.

The column vectors of V are the 'Right Singular Vectors' of **A**.

The SVD can be performed on matrices $A \in R^{m \times n}$, where $m \geq n$ (It can also be performed if $m < n$, but this is not interesting in the context

of 3D Computer Vision). In the case that $m = n$ there will be only non-zero positive diagonal elements. In the case that $m > n$, $s_1, \ldots, s_n$ are non-zero positive, $s_n+1, \ldots, s_m$ are zero. The SVD can be performed, such that the diagonal values of S are descending i.e. $s_1 \geq s_2 \geq \ldots \geq s_n \geq 0$.

The diagonal values of S are the square roots of the eigenvalues of $\mathbf{A^TA}$ and $\mathbf{AA}^T$ (hence the non-negativity of the elements of **S**).

The left singular vectors $\mathbf{u}_i$ are eigenvectors of $\mathbf{A}^T\mathbf{A}$.

The right singular vectors $\mathbf{v}_i$ are eigenvectors of $\mathbf{AA}^T$.

For those who want to know even more properties of the SVD, I will shortly introduce them.

- The SVD explicitly constructs orthonormal bases for the null-space and the range of a matrix.
- U, S, V provide a real-valued matrix factorization of M, i.e., $M = USV^T$.
- U is an n × m matrix with orthonormal columns, $U^TU = I_m$, where I_m is the m× m identity matrix.
- V is an orthonormal k × k matrix, $V^T = V^{-1}$.
- S is a n×n diagonal matrix, with the non-negative *singular values*, $s_1, s_2, \ldots, s_n$, on the diagonal.
- By convention the singular values are given in the sorted order $s_1 \geq s_2 \geq \ldots \geq s_n \geq 0$.
- The rank of M is given by the number of singular values s_j that are non-zero.
- The singular values $s_1, s_2, \ldots, s_n$ are unique; however, the matrices *U and V* are not unique.

The important inherent properties of SVD from the view point of image processing applications which makes it popular to use are:

- Singular Values (*Sv*s) are stable; i.e., any change to it doesn't affect the image quality.
- *Sv*s are able to represent inherent algebra properties of digital image.
- SVD preserves both one-way and non-symmetric properties which are not available by using DCT or DFT transformations.
- The size of matrices can be square or rectangular in SVD.
- *Sv*s are known to be invariant to some common attacks such as JPEG compression, noise addition, low pass filter (LPF), rotation, scaling and cropping.

COMPUTING SVD

The process of computing the SVD starts with a matrix (any image can be represented as a matrix). The matrix can be any size $n \times m$. An example would be:

$$A = \begin{pmatrix} 3 & 14 & 2 \\ 3 & 31 & 0 \\ 4 & 10 & 73 \end{pmatrix}$$

The first step in the SVD involves finding the transpose of the matrix A.

Definition 1: The transpose of an $m \times n$ matrix A is the $n \times m$ matrix A^T, whose columns are formed from the corresponding rows of A. The transpose of the example is:

$$A^T = \begin{pmatrix} 3 & 3 & 4 \\ 14 & 31 & 10 \\ 2 & 0 & 73 \end{pmatrix}$$

Definition 2: A 'symmetric matrix' is a matrix whose transpose is equal to the original matrix, i.e., $A^T = A$.

The second step in computing the SVD is to take the product of A^TA.

Definition 3: A dot product of two vectors is found by taking the sum of the products of the corresponding individual elements of the two vectors. Let

$$a = \begin{pmatrix} a_1 & a_2 & \dots & a_n \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ . \\ . \\ . \\ b_n \end{pmatrix}$$

The dot product of a and b is

$$a.b = a_1b_1 + a_2b_2 \dots + a_nb_n$$

Definition 4: A product (or matrix product) of the $m \times k$ matrix A and the $k \times n$ matrix B is the $m \times n$ matrix AB, whose $Ijth$ entry is the dot product of the vectors A_i and B_j where A_i is the ith row of A and B_j is the jth column of B:

$$(AB)_{ij} = a_{i1}\, b_{1j} + a_{i2}\, b_{2j} + \dots + a_{ik}\, b_{kj}$$

The example yields:

$$A^T A = \begin{pmatrix} 34 & 175 & 298 \\ 175 & 1257 & 758 \\ 298 & 758 & 5333 \end{pmatrix}$$

The next step requires finding the eigenvectors and eigenvalues.

Definition 5: An 'eigenvector' of an $n \times n$ matrix A is a nonzero vector x such that $Ax = \lambda x$ for some scalarλ. A scalar λ is called an *eigenvalue* of A if there is a nontrivial solution x of $Ax = \lambda x$, i.e., if lambda has an eigenvector.

Note that a scalar λ is an eigenvalue of A if and only if the relation $(A-\lambda I)\, x = 0$ has a nontrivial solution.

The eigenvalues from the example above (A^TA) are $\lambda_1 = 5488.69$, $\lambda_2 = 1133.3$, and $\lambda_3 = 2.00893$. The related eigenvectors from the example above (A^TA) are,

$$vec_1 = \begin{pmatrix} 0.0605 \\ 0.1816 \\ 1 \end{pmatrix}, vec_2 = \begin{pmatrix} -0.5749 \\ -5.3145 \\ 1 \end{pmatrix}, vec_3 = \begin{pmatrix} 25.3412 \\ -2.9297 \\ -1 \end{pmatrix}$$

Definition 6: The 'norm', or length, of vector v is the nonnegative scalar $\|v\|$ which is defined by

$$\|v\| = \sqrt{v.v} = \sqrt{v_1^2 + v_2^2 + \dots + v_n^2}$$

The norm of the eigenvectors from the example is:

$$|| \, vec_1 \, || = \sqrt{|0.0605|^2 + |0.1816|^2 + |1|^2}$$

$$= \sqrt{1.0366} = 1.01816$$

$$\|vec_2\| = 5.43821, \ and \ \|vec_3\| = 25.5396 .$$

Definition 7: A 'unit vector' is a vector of length one. To *normalize* a nonzero vector is to divide it by its length. Observe that a normalized vector is a unit vector. The normalized eigenvectors from the example are:

$$v_1^T = \frac{vec_1^T}{|vec_1|} = \frac{\begin{pmatrix} 0.0605 & 0.1816 & 1 \end{pmatrix}^T}{0.01816}$$

$$= \begin{pmatrix} 0.5938 & 0.1784 & 0.9822 \end{pmatrix}^T$$

$$v_2^T = \frac{vec_2^T}{|vec_2|} = \begin{pmatrix} -0.1057 & -0.9772 & 0.1839 \end{pmatrix}^T,$$

and

$$v_3^T = \frac{vec_3^T}{|vec_3|} = \begin{pmatrix} 0.9926 & -0.1148 & -0.0392 \end{pmatrix}^T$$

Definition 8: A real 'positive semi definite matrix *A'* is a real symmetric matrix for which the eigenvalues are all non-negative.

Proposition 1: For any *A*, A^TA is positive semi-definite.

Definition 9: The 'singular values' of matrix *A* are the square roots of the eigenvalues of the positive semi-definite matrix A^TA and are denoted by $s_1,\ldots,s_n$, arranged in non-increasing order. That is,

$$s_1 = \sqrt{\lambda_1} \geq s_2 = \sqrt{\lambda_2} \geq \ldots \geq s_n = \sqrt{\lambda_n},$$

where λI *I* is the *ith* eigenvalue of A^TA.

Note, because A^TA must be a positive semi-definite matrix, the singular values of *A* are real numbers. The singular values for the example are: s_1=74.0857, s_2= 33.6646 and s_3=1.4174.

Definition 10: Two vectors *u* and *v* in R^n are orthogonal to each other if $u.v = 0$.

Definition 11: A set $\{u_1, u_2, \ldots, u_n\}$ is an orthonormal set if it is an orthogonal set of unit vectors.

Let the right singular vectors be the columns of the eigenvector matrix *V*.

V is the transpose of v_1, v_2, and v_3.

$$V = \begin{pmatrix} 0.0594 & -0.1057 & 0.9926 \\ 0.1784 & -0.9772 & -0.1148 \\ 0.9822 & 0.1839 & -0.0392 \end{pmatrix}$$

Definition 12: The 'column space' of $m \times n$ matrix *A*, written as Col *A*, is the set of all linear combinations of the columns of *A*. If $A = \{a_1, \ldots, a_n\}$, then Col *A* = Span $\{a_1, \ldots, a_n\}$. The 'rank' of *A*, denoted by rank *A*, is the dimension of the column space of *A*.

Definition 13: The left singular vectors can now be found by taking $u_i = \frac{1}{s_i} A v_i$ when $s_i = 0$.

For example:

$$u_1 = \frac{1}{s_1} A v_1$$

$$= \frac{1}{74.0857} \begin{pmatrix} 3 & 14 & 2 \\ 3 & 31 & 0 \\ 4 & 10 & 73 \end{pmatrix} \begin{pmatrix} 0.0594 \\ 0.1784 \\ 0.9822 \end{pmatrix} = \begin{pmatrix} 0.0626 \\ 0.7705 \\ 0.9951 \end{pmatrix}$$

$$u_2 = \frac{1}{s_2} A v_2 = \begin{pmatrix} -0.4049 \\ -0.9093 \\ 0.0959 \end{pmatrix}$$

$$u_3 = \frac{1}{s_3} A v_3 = \begin{pmatrix} 0.9122 \\ -0.4089 \\ -0.0258 \end{pmatrix}$$

Let *U* be the matrix of left singular vectors. In this example, this gives the matrix

$$U = \begin{pmatrix} 0.0626 & -0.4049 & 0.9122 \\ 0.0770 & -0.9093 & -0.4089 \\ 0.9951 & 0.0959 & -0.0258 \end{pmatrix}$$

To check the stability of the singular values, an experiment was conducted on 8 bit gray scale 512× 512 Lena image. In this experiment, original singular values were compared with singular values after applying various attacks on them. Table 1 shows the first four singular values of

Table 1. Various attacks on Lena image, its singular values

Image	S_1	S_2	S_3	S_4
Original Image	151.5234	42.2745	36.1516	27.9067
JPEG Compression(Q=20)	151.6007	42.2129	36.0787	27.6894
Rotation(15 Degree)	144.1636	48.0665	39.9409	28.7351
Scaling (512-256-512)	152.1418	42.1731	36.0141	27.7552
Scaling (512-1024-512)	152.7299	42.2633	36.1170	27.8758
Gaussian Noise (M=0, V= 0.01) Salt & Paper Noise (M=0 V=0.01)	158.5279 152.3987	40.7767 41.9533	35.4015 35.8831	27.3755 27.7077
Median Filter {3X3}	151.2235	42.2745	36.1516	27.9067
Histogram Equalization	151.5234	42.2745	36.1516	27.9067

the original and modified image after applying various attacks. The singular values do not change very much.

SVD IMAGE PROPERTIES

SVD is robust and reliable orthogonal matrix decomposition method. Due to SVD conceptual and stability reasons, it becomes more and more popular in 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 needs more investigation and contribution. Several SVD properties are highly advantageous for images such as; its maximum energy packing, solving of least squares problem, computing pseudo-inverse of a matrix and multivariate analysis. A key property of SVD is its relation to the rank of a matrix and its ability to approximate matrices of a given rank. Digital images are often represented by low rank matrices, and therefore able to be described by a sum of a relatively small set of Eigen images. This concept rises the manipulating of the signal as two distinct subspaces. For a complete review, the theoretical SVD related theorems are summarized in the follows (Sadek, 2012).

- **SVD Subspaces:** SVD is constituted from two orthogonal dominant and subdominant subspaces. This corresponds to divide the M-dimensional vector space into dominant and subdominant subspaces. This attractive property of SVD is utilized in noise filtering and watermarking.
- **SVD Architecture:** For SVD decomposition of an image, singular value (*Svs*) specifies the luminance of an image layer while the corresponding pair singular vectors (SCs) specify the geometry of the image layer. The largest object components found in the image by using the SVD generally correspond to Eigen images associated with the largest singular values, while image noise corresponds to Eigen images associated with the *Sv*s.

- **PCA vs. SVD:** Principle component analysis (PCA) is also called the Karhunen-Loéve transform (KLT) or the Hotelling transform. PCA is used to compute the dominant vectors representing a given data set and provide an optimal basis for minimum mean squared reconstruction of the given data. The computational basis of PCA is the calculation of the SVD of the data matrix, or equivalently the eigenvalues decomposition of the data covariance matrix.
- **SVD Multiresolution:** SVD has the maximum energy packing among the other transforms. In many applications, it is useful to obtain a statistical characterization of an image at several resolutions. SVD decomposes a matrix into orthogonal components with which optimal sub rank approximations may be obtained. With the multi-resolution SVD, the following important characteristics of an image may be measured, at each of the several level of resolution: isotropy, specify of principal components, self-similarity under scaling, and resolution of the mean squared error into meaningful components.
- **SVD Oriented Energy:** In SVD analysis of oriented energy both rank of the problem and signal space orientation can be determined. SVD is a stable and effective method to split the system into a set of linearly independent components, each of them bearing its own energy contribution. SVD is represented as a linear combination of its principle components, a few dominate components are bearing the rank of the observed system and can be severely reduced. The oriented energy concept is an effective tool to separate signals from different sources, or to select signal subspaces of maximal signal activity and integrity. Recall that the singular values represent the square root of the energy in corresponding principal direction. The dominant direction could equal to the first singular vector V_1 from the SVD decomposition. Accuracy of dominance of the estimate could be measured by obtaining the difference or normalized difference between the first two *Sv*s.

WATERMARK

The idea of using singular values for watermarking was explored a few years ago by Liu and Tan (Liu, R., Tan, T., 2002). The main idea of this approach is to find the SVD of an original image and then modify its singular values to embed the watermark.

Digital watermarking technique is one of most important methods in information hiding and IPR (Intelligence Properties Right) protection and authentication. It is the process of embedding information into a digital signal in a way that is difficult to remove. The process of embedding a certain piece of information (technically known as watermark) into multimedia content including text documents, images, audio or video streams, based on which the watermark can be detected or extracted later to make an assertion about the data.

In visible digital watermarking, the information is visible in the picture or video. Typically, the information is text or a logo, which identifies the owner of the media. When a television broadcaster adds its logo to the corner of transmitted video, this also is a visible watermark.

In invisible digital watermarking, information is added as digital data to audio, picture, or video, but it cannot be perceived as such (although it may be possible to detect that some amount of information is hidden in the signal).

WATERMARK PROPERTIES

In general, a digital watermark should have several different properties. The most important are imperceptibility, robustness and security. Imperceptibility means that the watermarked data should be perceptually equivalent to the original, un-watermarked data.

In some applications, the watermark may be perceptible as long as it is not annoying or obtrusive; however, many applications require that the watermark be imperceptible. Security means that unauthorized parties should not be able to detect or manipulate the watermark. Cryptographic methods are typically employed to make watermarks secure.

Watermarking means embedding a piece of information into a multimedia content, such as a video, an audio or an image in such a way that it is imperceptible to a human observer, but easily detectable by a computer. Before the emergence of digital image watermarking, it was difficult to achieve copyright protection, authentication and data hiding, but now it is easy to achieve these goals by using watermarking techniques. Every watermarking algorithm consists of an embedding algorithm and a detection algorithm.

Embedded watermarks may have several properties such as robustness, fidelity, and tamper-resistance. The robustness means that the watermark must be robust to transformations that include common signal distortions such as digital-to-analogue conversion, analogue-to-digital conversion, and lossy compression. Fidelity means that the watermark should be neither noticeable to the viewer nor degrading for the quality of the content. Tamper-resistance means that the watermark is often required to be resistant to signal processing algorithms. These properties depend on the application. The watermark can be embedded in the spatial domain or in a transform domain.

The SVD mathematical technique provides an elegant way for extracting algebraic features from an image. The main properties of the *Sv*s matrix of an image can be exploited in image watermarking. This matrix has a good stability. When a small perturbation occurs in an image, the variation of its *Sv*s can be neglected. Using this property of the *Sv*s matrix of an image, the watermark can be embedded to this matrix without a large variation in the obtained image.

Digital watermarking is a recent method of protecting digital multimedia data (audio, image and video) against unauthorized copying. A digital watermark is a signal added to the original signal, which can later be extracted or detected. The watermark is intended to be permanently embedded into the digital data so that authorized users can easily access it. At the same time, the watermark should not degrade the quality of the digital data.

The main driving force is the concern over protecting copyright: since audio, video and other works become available in digital form, the ease with which perfect copies can be made may lead to large-scale unauthorized copying, and this is of great concern to music, film, book and software publishing industries. At the same time, moves by various governments to restrict the availability of encryption services have motivated people to study methods by which private messages can be embedded in seemingly innocuous cover messages.

AUDIO WATERMARK

Before applying the SVD to an audio signal, the issue of how to organize the data into matrix form has to be addressed. The audio signal should first be split into frames whose length is denoted as *len*. The magnitude spectrum of the signal in each frame can then be computed. This spectrum will contain *len/2* frequency bins below the Nyquist frequency. If we put all of the frequency components of several consecutive frames into the same matrix, there should be redundancy, as generally, each component's

magnitude would be similar to its value in the previous and the next frame. An alternative approach to organizing the Reduced Singular Value Decomposition RSVD input matrix is to put all the spectral components from one frame into a single matrix. For example, with a frame length of 1024 we just use the first 512 magnitude values and transform them into a 64×8 matrix. The magnitudes of the first 64 frequency bins will be put into the first column, the second 64 bin magnitudes into the second column and so on. The values in the first column will normally but not always be more significant than those in subsequent columns (Wang et al, 2010).

The steps of the SVD audio watermark embedding algorithm are summarized as follows:

1. The 1-D audio signal is transformed into a 2-D matrix (**A** matrix).
2. The SVD is performed on the **A** matrix.

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}}$$

3. The chaotic encrypted watermark (**W** matrix) is added to the *Sv*s of the original matrix.

$$\mathbf{D} = \mathbf{S} + k\mathbf{W}.$$

A small value of k of about 0.01 is required to keep the audio signal undistorted.

4. The SVD is performed on the new modified matrix (**D** matrix).

$$\mathbf{D} = \mathrm{U}_{\mathrm{w}}\, \mathrm{S}_{\mathrm{w}}\, \mathrm{V}^{\mathrm{T}}{}_{\mathrm{w}}\, .$$

5. The watermarked signal in 2-D format (**A***w* matrix) is obtained using the modified matrix of *Sv*s (**S***w* matrix).

$$\mathbf{A}_{\mathrm{w}} = \mathbf{U}\, \mathbf{S}_{\mathrm{w}}\mathbf{V}^{\mathrm{T}}.$$

6. The 2-D $\mathbf{A}_{\mathrm{w}}$ matrix is transformed again into a 1-D audio signal.

To extract the possibly corrupted watermark from the possibly distorted watermarked audio signal, given $\mathbf{U}_{\mathrm{w}}$, **S**, $\mathbf{V}_{\mathrm{w}}$ matrices, and the possibly distorted audio signal, the above steps are reversed as follows:

1. The 1-D audio signal is transformed into a 2-D matrix $\mathbf{A}_{\mathrm{w}}$. The * refers to the corruption due to attacks.
2. The SVD is performed on the possibly distorted watermarked image (**A**w matrix).

$$\mathbf{A}^*_{\mathrm{w}} = \mathbf{U}^* \mathbf{S}^*_{\mathrm{w}} \mathbf{V}^{*\mathrm{T}}\, .$$

3. The matrix that includes the watermark is computed.

$$\mathbf{D}^* = \mathbf{U}_{\mathrm{w}}\mathbf{S}_{\mathrm{w}}\mathbf{V}^{\mathrm{T}}{}_{\mathrm{w}}\, .$$

4. The possibly corrupted encrypted watermark is obtained.

$$\mathbf{W}^* = (\mathbf{D}^* - \mathbf{S})/k.$$

5. The obtained matrix **W*** is decrypted.
6. The correlation coefficient between the decrypted matrix and the original watermark is estimated. If this coefficient is higher than a certain threshold, the watermark is present.

IMAGE WATERMARKING

In this section we present general algorithm to embed a gray scale image into another gray scale image of the same size using SVD (the same method for color image can apply for each color component Red, Green and Blue, and then later combine together as one image). Let the matrix **A**, with elements aij, $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$, represent the host image which needs to be watermarked. Let **W** represent the matrix of the image to be embedded. As a first step, we compute the SVD of both **A** and **W** (Agarwal & Santhanam, 2008).

$\mathbf{A = Ua\ Sa\ V_a^T}$

For the watermark image

$\mathbf{W = Uw\ Sw\ V_w^T}$

Now, we add the scaled eigenvector **Vw** of watermark to that of the original image,

$\mathbf{V = Va + \propto Vw}$

where $\propto$ is the scaling factor, typically, $0 \leq \propto \leq 1$, so that the intensity of the watermark **W** is less compared to the original image **A**. Note that, within the framework of SVD, $\mathbf{Vw\ V_w^T = I}$, where **I** is the identity matrix. Similar relation holds good for **Va** too. As $\propto \rightarrow 0$, the approximation that **V** is an orthogonal matrix, i.e., $\mathbf{VV^T \approx I}$ gets better. This property is important in the next step for constructing the watermarked image. We get the watermarked image as,

$\mathbf{Ac = Aa\ V^T}$.

Also, a modified to this algorithm, by apply the SVD to each component of the original image (A) (red, green and blue component)

SVD (A) = $U_i\ S_i\ V_i$ i= 1, 2, and 3. i= 1 for red component, 2 for green component and 3 for blue component.

We can perform the embedding procedure by the following steps.

$$S_{wj} = S_i + \propto T_i$$

$\propto$ is a scaling factor, which controls the strength of the watermark to be inserted. Because the embedded watermark can't degrade the host image quality, the value of $\propto$ should be small. During the process of SVD, three matrices U, V and S are produced.

(U1, V1); (U2, V2); (U3, V3) these matrices are the user's secret keys, which don't contain any information about three watermarks.

A method of desirable watermarking should satisfy the ownership of imperceptibility, where the integrated watermark is not visible for the observer. (A) Represent the Host image.

(T1, T2, T3) Represent the three watermarks for the protection of copyright, (A1,A2,A3) represent the three components of RGB image, (B1, B2, B3) represent the three components watermarked of RGB image, (B) The watermarked image.

Here, we define the value of scaling factor $\propto = 0.1$ and inject these watermarks into the RGB components of the host image (EL Gorfte et al, 2013).

VIDEO WATERMARKING

The watermark is a digital code embedded in the multimedia (audio, images, video, etc.) before transmission or broadcasting, which typically indicates the copyright owner. If different watermarks are appending to individual copies of the video, watermarking can be also used to indicate the identity of the legal receiver of each copy. This allows tracing back an illegally reproduced copy to the receiver of the copy from which the illegal copy is originated. As an important branch of watermarking algorithms, video watermarking is attracting more and more attention. In video watermarking, watermark can be embedded in the spatial and/or transform domains.

The embedding algorithm is based on transforming the host video using the SVD operator and then embedding the watermark information in the S, U, or V matrices diagonal-wise. We described in details in the following steps (Rajab et al, 2009):

Step 1: Divide the video clip into video scenes ***Vsi.***
Step 2: Process the frames of each video scene using SVD described in steps 3 - 9 below.

Step 3: Convert every video frame ***F*** from RGB to YCbCr color space.

Step 4: Compute the SVD for the Y matrix for each frame ***F***. This operation generates 3Matrices (U, S, V) such as: $\boldsymbol{Y = U_Y S_Y V_Y}$

Step 5: Rescale the watermark image so that the size, of the watermark will match the size of the matrix which will be used for embedding either U, V or S.

Step 6: Embedding can be done in one of the three SVD matrices: *U, V, or S*, as follows:

Embedding in Matrix U Diagonal-Wise

1. Inverse each diagonal value ($u_{i,i}$) in The U matrix, such that $x = 1/ u_{i,i}$
2. Embed the binary bits of the watermark $\boldsymbol{W_{Vsi}}$ into the integer part of x by substituting the watermark bit

$\boldsymbol{W_i}$ with the 7th bit of $\boldsymbol{x}$.

3. Apply the inverse to each x, to get the modified values of U matrix, such that $u_{i,i}' = 1/x'$.
4. Apply inverse SVD on the modified coefficient matrix *U '*. Such as:

$$\mathbf{Y' = U_Y' \, S_Y V_Y^T}$$

Embedding in Matrix V Diagonal-Wise

1. Inverse each diagonal value ($v_{i,i}$) in The V matrix, such tha $x = 1/ v_{i,i}$
2. Embed the binary bits of the watermark W_{Vsi} into the integer part of x by substituting the watermark bit

$\boldsymbol{W_i}$ with the 7th bit of $\boldsymbol{x}$.

3. Apply the inverse to each x, to get the modified values of V matrix, such that $v_{i,i}' = 1/x'$.
4. Apply inverse SVD on the modified coefficient matrix V '. Such as:

$$\mathbf{Y' = U_Y S_Y V_Y'^T}$$

Embedding in Matrix S Diagonal-Wise

1. Embed the binary bits of the watermark $\boldsymbol{W_{Vsi}}$ into the integer part of each diagonal value of the S matrix $s_{i,i}$ by substituting the watermark bit $\boldsymbol{W_i}$ with the7th bit of $s_{i,i}$
2. Apply inverse SVD on the modified coefficient matrix S ' such as:

$$\mathbf{Y' = U_Y S_Y' \, V_Y^T}$$

where Y' is the updated luminance in the YCbCr color representation. This operation produces the final watermarked video frame ***F '***.

Step 7: Convert the video frames ***F'*** from YCbCr to RGB color space.

Step 8: Reconstruct frames into the final watermarked video scene $\boldsymbol{V_{si}'}$.

Step 9: Reconstruct watermarked scenes to get the final watermarked video clip.

HYBRID SVD

The schemes which are applied with or after cascading of any transform domain are called hybrid SVD based schemes. DCT, DWT, FFT are few most popular frequency domains which used with SVD to make watermarking schemes more robust.

SVD and DCT Based Algorithm

There are various types of hybrid watermarking schemes based on DCT and SVD has been proposed. In general in case of pure DCT based watermarking schemes the DCT transformation is applied to the original image and then frequency

coefficients from lowest to highest are mapped in zig-zag sequence into some forms of quadrants or blocks. These DCT coefficients are modified to embed watermark. Whereas in case of SVD based watermarking SVD transformation is applied to the entire image and then the singular values of the image are modified to embed the watermark in host image. In hybrid DCT-SVD watermarking schemes both DCT & SVD features are combined, that means DCT transform is applied to the cover image and also to the watermark. DCT coefficients of cover image are mapped to some quadrants using zig-zag sequence. SVD is applied to each quadrant and also to the DCT coefficients of watermark, then the singular values of each quadrant modified with the singular values of DCT coefficients of watermark. The hybrid DCT-SVD based watermarking scheme when embedding watermark in lowest frequency shows robustness to some set of attacks, while embedding in higher frequency shows robustness to another set of attacks. In general a hybrid DCT-SVD watermarking scheme shows robustness against a set of attacks like Gaussian blur, Gaussian noise, JPEG compression, rescaling cropping, histogram equalization etc. but shows less robustness to rotation and translation operation.

SVD and DWT Based Algorithm

As in case of DCT, here in DWT the cover image is decomposed into four sub-bands i.e. LL, LH, HL, and HH. In hybrid SVD-DWT watermarking schemes, SVD is applied to the sub-bands and also to the watermark, then the singular values of the sub-bands are modified by embedding the singular values of watermark. Finally four sets of DWT coefficients are obtained and applying the inverse DWT using the modified DWT coefficients, watermarked image is being produced. Same as in DCT-SVD, watermarking scheme embedding watermark in various sub-band shows robustness to different kinds of set of attacks. In general hybrid DWT-SVD watermarking scheme robust to a set of attacks including Gaussian blur, Gaussian noise, JPEG compression, JPEG 2000, compression, rescaling, cropping etc. But shows less robustness against sharpening, rotation, contrasting and histogram equalization, also when embedding watermark in LL band although the extracted watermark is best in visual quality but after embedding of watermark degrades the image quality to some extents.

STEGANOGRAPHY

The amazing developments in the field of network communications during the past years have created a great requirement for secure image transmission over the Internet. Steganography is a good solution to transmit the message in secure base, also it is possible to increase the security of steganography by combine it with encryption. Steganography is the science and art of hiding data of digital medium in other digital medium called 'cover object' in such a way that the existence of the message is concealed. The cover object along with the hidden message is known as the 'stego object' or 'steganogram'. Steganography is in contrast to cryptography where the existence of the hidden message is known, but the content is intentionally obscured.

There are several properties important in the creation and evaluation of an effective stego-algorithm. These include the following:

- **Capacity:** Measures the amount of payload that can be embedded in a fixed size of cover file. It is measured in bits (of payload) per byte (of cover). For example, in the LSB embedding scheme, the sender chooses as a cover image a gray scale, bit-mapped image, and replaces the least significant bit of each pixel with one bit of payload. Assuming each pixel occupies 8 bits, this scheme has a capacity of 1 bit/byte.

- **Perceptibility:** Describes the ability of a third party (not the intended recipient) to visually detect the presence of hidden information in the stego image (or audibly, in the case of audio data). Note that we do not require the third party to extract the information, just perceive its existence. We say that the steganography embedding algorithm is *imperceptible* when used on a particular image if an innocent third party interested in the content of the cover image, is unaware of the existence of the payload. Essentially this requires that the embedding process not degrade the visual quality of the cover image.
- **Detectability:** Describes the probability that a determined adversary, who suspects steganography, will be able to determine the existence of a payload, thus compromising the message's security. In other words, a stego algorithm provides low security if the payload is detectable in the stego image with a high probability.
- **Undetectability:** Obviously a much more difficult requirement to meet than is imperceptibility, but as the use of steganography increases, so will the use of steganalysis is. The concept of a 'safe bit-rate' is related to detectability of a stego algorithm. The safe bit-rate (SBR) is the maximum capacity of a stego algorithm when applied to a particular image that is not detectable by steganalysis. The SBR is therefore dependent not only on the algorithm used to embed the data, but the data itself as well as the steganalysis techniques available to detect it.
- **Robustness:** Characterizes the ability of the payload to survive the embedding and extraction process, even in the face of manipulations of the stego image such as filtering, cropping, rotating and compression.
- **Speed:** Reflects the computational effort required to embed and extract the hidden data. It is well-understood that there is always a tradeoff between capacity and visual imperceptibility, and capacity and detectability (Bergman & Davidson, 2005).

The message to be embedded will begin as a string of binary values $b_1b_2b_3$...where $b_i \in \{0,1\}$. Next, these will need to be converted to signed bits using a transformation using either of the transformations $\left(b \rightarrow (-1)^b \quad or \quad b \rightarrow 2b \rightarrow -1\right)$, making a message $p_1 p_2 p_3 \ldots$ where $pi \in \{-1,1\}$. The basic idea then is to embed the signed message into the image by changing the signs of certain entries in the matrix U (from the SVD of a matrix that describes part of an image) to correspond with the signed bit values $\{pi\}$. Each pixel of a gray scale image (cover) M, is a value in the range 0 to 255. To expand this to a colored image in a RGB format, each pixel would have three values in that range, ideally allowing the embedding of three times as many values.

The cover image is divided into a series of $n \times n$ blocks in some standard order.

Message bits are embedded into A by a simple four-step process:

1. Compute the SVD ($U\ S\ V^T$) of A.
2. Transform U to U'
   a. Set certain components $u'ij = p_k \cdot |u_{ij}|$, where k the next bit from message.
   b. Chose remaining components to ensure that U' is still orthogonal.

The visual quality of A is primarily determined by its largest singular values and singular vectors. By assumption, those are the left-most values of S and the left-most columns of U and V. Part of this strategy is to leave those columns untouched in order to achieve imperceptibility in this method.

3. Compute $A' = U' S V^T$.
4. Clip and round the entries in A' to integers in the range 0…255. The resulting matrix A_E will be a block of stego image.

Extraction Algorithm A_E denotes the stego image and *SVD* is applied to retrieve the hidden message.

1. Compute the SVD_E, $U_E S_E V_E^T$ of A_E.
2. Extract payload bits from the signs of the entries in the triangular portion of U_E: $P_k = u_{Eij} / |u_{Eij}|$

For example: to embed the message: 0 0 1 1 0 1011 …

- Keep first 2 columns unchanged.
- Construct UE by changing {uk2}, (see Figure 2):

$$u_{(E)k2} = \begin{cases} +|u_{k2}| & if\ message\ bit\ is\ 1 \\ -|u_{k2}| & if\ message\ bit\ is\ 0 \end{cases}$$

- Last entries z_{72} and z_{82}: calculated so that column 3 is orthogonal to columns 1 & 2.
- Next column: insert 4 bits; last 3 entries calculated so that column 4 is orthogonal to columns 1, 2 & 3.
- Etc.
- Reconstruct the block with the embedded bits:

A_E = clip (round (U_E S V^T)

where U_E is the matrix constructed in previous, *S, V* are from original matrix (image) *A*.

- **Round:** Rounds to nearest integer; *clip*: values less than 0 get reset to 0; values larger than 255 get reset to 255.
- Do this for each block.

- Capacity for 8 × 8 block with two columns protected: 15 bits per 64 pixels = .234 bits/pixel bit embedding rate.
- 0.2 – 0.5 bits/pixel is typical for "good" stego algorithms. May not be "safe enough" to avoid detection.

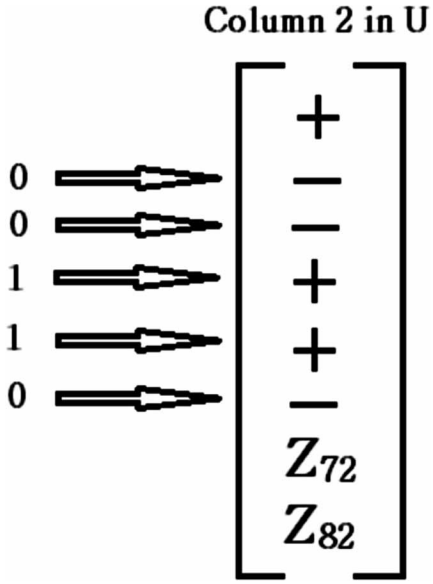
With this algorithm, the error between the extracted message and the embedded message is typically 8% - 10% of the total bits embedded.

Most of the error comes from the fact that the change from *U* to U_E is large relative to the magnitudes of the entries in *U*. There are several solutions to reduce error.

STEGANALYSIS

Steganalysis is the art of discovering the very presence of hidden data in cover objects. Steganalysis can be broadly classified into two groups: algorithm specific (targeted) methods and universal (blind) methods. Algorithm specific steganalysis assumes that the steganographic method is known by the attacker. The attacker takes advantage of

Figure 2. How to embed a message in a column



this prior knowledge to design methods to reveal the existence of the hidden data. The short coming of this type of steganalysis is that their satisfactory performance is restricted to a specific steganography. Universal steganalysis methods aim to overcome this problem. Instead of using any *a priori* information, they take into account all available steganography methods to devise a single steganalysis framework. It is supposed that a blind steganalysis method can detect any steganography if sufficient numbers of cover and stego images have been taken into account during the design process.

All universal steganalysis methods assumed that data-hiding destroys the underlined statistics of natural images. Therefore, a common characterization should be possible if the features incorporated to the classification process are sensitive to the embedding noise and insensitive to the image content. Universal steganalysis schemes can be divided into two categories: spatial domain and discrete cosine transform (DCT) domain. The methods belonging to the former one extract the features from the pixel information, while the latter ones attack on the DCT coefficients.

The SVD steganlysis method attacks steganographic content using the features derived from singular values. Let us assume a full rank matrix.

If any two rows or columns of this matrix are modified so that they become linearly dependent, it can be observed that the lowest singular value vanishes. If this process is repeated using the next row or column, the second lowest singular value becomes zero. This observation comes up with two main ideas which are the pillars of this method.

First, the reaction to the changes on the matrix content starts from the lowest singular value. Second, the lower valued singular values closeness to zero indicates a group of vectors closeness to the linear dependency. This observation can be used to model the soft relationship between the image rows and columns which will be disturbed by the embedding process.

Due to the aforementioned unequal effect of embedding noise on the singular values, it is necessary to adopt a function which can intensify the lower valued singular values for a powerful steganalysis and can attenuate the higher valued ones to normalize the different energy levels of different images. For this purpose, a function comprising the logarithm of the inverse power of singular values ($\log(s_x^{-1})$) is devised to derive features for the steganalysis. Since spatial domain represents a strong dependency between the pixels in the local neighborhoods the features are extracted from the sub-blocks representing the locality in the spatial domain rather than the entire image. It is obvious that when the block size increases the number of examined blocks decreases. This results in decreasing the dependencies which are considered. To alleviate this problem, the sub-blocks are overlapped proportionally to the block size in order to be able to take into account the correlations within and among sub-blocks. Consequently, the feature extraction algorithm is described as follows.

Step 1: Divide image A, into sub-blocks of size W×W, where W= 3,4,…,27 according to the following overlapping rules:

If W < 8, no overlapping

If 8<= W <= 13, 50% overlapping

If W>13, 75% overlapping

Step 2: For each particular, calculate the singular value vector S_v of each sub-block j

SVD (sub-block$_j$) = $S_{vj} = (s_{1j}, s_{2j}, \ldots\ldots s_{wj})$

Step 3: Calculate the natural logarithm of the inverse power of each singular value and add the singular values up with respect to the related sub-block j

$$SvB_j = \sum_{i=1}^{w} \log\left(s_{ij}^{-1}\right),\ j = 1,2,\ldots T_w,\ \ s_{ij} \neq 0$$

where T_w is the total number of sub-blocks sizes of W×W

Step 4: Sum the final results obtained in Step 3 and normalize them with the number of total sub-blocks

$$F_w = \frac{1}{T_w}\sum_{j=1}^{Tw} S_v B_j,\ \ W = 3,\ldots,27\,.$$

Using this algorithm, we obtain 25 dimensional (25D) features for each image.

IMAGE ENCRYPTION

The field of encryption is becoming very important in the present era in which information security is of utmost concern. Security is an important issue in communication and storage of images, and encryption is one of the ways to ensure security. Image encryption has applications in internet communication, multimedia systems, medical imaging, telemedicine, military communication, etc.

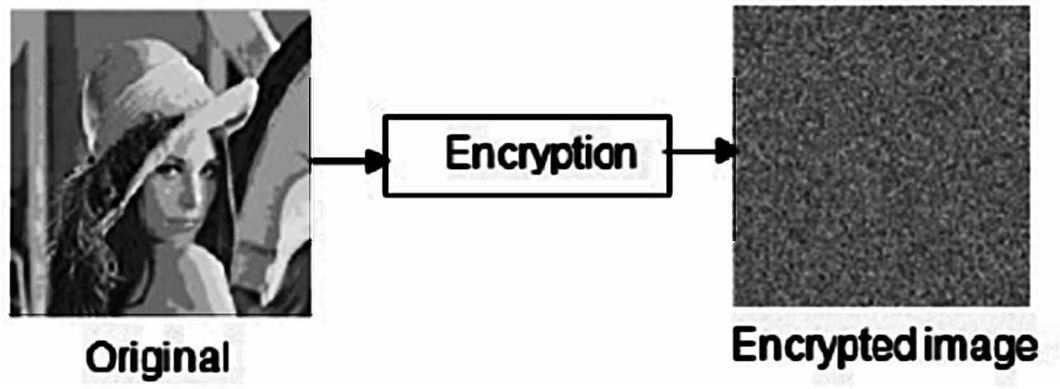
Image information is different from the text data, it has larger amount of data, higher redundancy and stronger correlation between pixels. Traditionally developed encryption algorithm such as RSA, DES is suitable for text encryption but not suitable for image encryption directly because of two reasons.

The first one, that the image size is almost always much greater than that of text. Therefore, the traditional cryptosystems need much time to directly encrypt the image data. The other problem is that the decrypted text must be equal to the original text. However, this requirement is not necessary for image data. Due to the characteristic of human perception, a decrypted image containing small distortion is usually acceptable, encrypted image shows in Figure 3.

At present there are many image encryption algorithms are available but these algorithms doesn't satisfy the requirement of modern cryptographic mechanism and they are prone to attacks. In the recent years, the image encryption has been developed to overcome the above disadvantages.

1. The first step in this work is to create the necessary keys for encryption image (A)
2. Scrambling the pixels value is the second step, scrambling the image values as follow:

Figure 3. Image encryption



$$A1 = Key1 * \max(A) - A, A2 = Key1 * \max(A1) - A1$$

3. Appling SVD for both matrices resulting from the previous step.

SVD (A1) = UA_1, SA_1, VA_1

SVD (A2) = UA_2, SA_2, VA_2

4. Rebuild new matrix from the results of SVD process in step 3, this can be done by replacing the singular values (*Sv*) of A1 with singular values of A2.

$C1 = UB_1 * SB_2 * VB_1^T$

$C2 = UB_2 * SB_1 * VB_2^T$

5. For more complexes, the same steps above can be repeated to create new matrices.

Scrambling the elements in matrices (C1, C2) to get new matrices (D1, D2)

$$D1 = C1 - C2, \quad D2 - KEY3 * D1 + C2$$

6. Then, SVD applied for both matrices (D1, D2) and replaces the singular values of D1 with singular value of D2 as we did in previous steps to create new matrices (E1, E2)

$[UD_1, SD_1, VD_1] = SVD$ (D1)

$[UD_2, SD_2, VD_2] = SVD$ (D2)

So,

$$E1 = UD1 * SD2 * VD1^T, \quad E2 = UD2 * SD1 * VD2^T$$

7. Combine (E1, and E2) in one matrix

$$F = \left[E1\,E2\right]$$

8. Finally rescaling F, by using the following relation

$$FF = \frac{F - MI}{MA - MI}$$

where, MA is the maximum number in matrix (F), and MI is the minimum number in the matrix (F).

The decryption process is the inverse of the above steps.

SVD PREVIOUS RESEARCH

Nelson (Nelson et al, 2009) focused on the Ghost Circuitry Detection based on SVD in their research.

Ghost Circuitry (GC) insertion is an intentional hardware alteration of the design specification and IC implementation. The alterations only affect the circuit's functionality in a few specific circumstances and are hidden otherwise. GC is more difficult to detect than design bugs or manufacturing faults, since it is intentionally implanted to be unperceivable by the current debugging and testing methodologies and tools. The vast number of possibilities for inserting GC further complicates detection.

In a GC insertion attack, the adversary adds one or more gates such that the functionality of the design is altered. The gates can be added so that no timing path between primary inputs and flip-flops (FFs) and primary outputs and FFs is altered. However, leakage power is always altered. Even if the attacker gates the added circuitry, the gating requires an additional gate.

Manufacturing variation in power and delay behavior of gates is modeled by associating each gate with a scaling factor, α, which multiplies both delay and leakage current.

Measurements of total leakage power and path delay for various circuit inputs gives rise to linear equations with the scaling factors as the unknowns. Each set of measurements produces

a linear system $Ga = m+e$ where a is the vector of scaling factors, also referred to as the α-values, and related to gate size.

- $m+e$is a vector of measured values
- m would be the measured value if there is no measurement error
- e is the measurement error associated with each measured value
- G is derived from the expected power and/or delay characteristics of the gates.

For Ng number of gates in the circuit and Nm number of measurements, G is $Nm \times Ng$, a is $Ng \times 1$, and m is $Nm \times 1$.

More abstractly, one can imagine the circuit's gate characteristics split into two components represented by G and a. G represents the characteristics of gate classes, i.e. 2-input NANDs power and delay characteristics for a given input vector, and it is inherent the circuit design. This information is readily available and in our experiments we have used the values provided by for delay and for leakage power.

The vector a, which is a vector of α-values for all the gates in the circuit, represents the unknowns in the equation. In other words, a is the fingerprint for the circuit just as the α-value is the fingerprint for the individual gate. Due to manufacturing variability, gate sizes are not exactly matched to the design specifications. The size of each gate in the circuit of each fabricated IC can have a variety of values. All circuits accordingly will have a large variety of sizes for most or all of their gates, and hence the extremely large combinations of possibility for a results in a unique fingerprint for each circuit. Splitting each manufactured circuit into an invariant and into a variant component results in, G, which is universal across all circuits of the same design for the same set of input vectors, and a, which represents the unique characteristics of the fabricated circuit.

A large set of measurements are taken for the total circuit. As we can only access the input and output pins of the circuit, all the measurements made, represented by $m+e$, are made from a global circuit or path level and not at the individual gate level. Obviously, if we were able to measure these values at the gate level, we would easily be able to solve for each gate's α-value.

We do consider error in the formulation, as measurement error is possible when measuring total leakage power for the circuit and total delay along a path of the circuit from input to output pin. This is represented by e, which is the error that may be introduced in the measurement for each input vector or pair of input vectors.

A singular value decomposition $G = USV^T$ is used in the following way. $G+$, the pseudo-inverse of G, gives a least-squares solution to the system, a', an approximation of the scaling factors given the possibility of measurement errors being introduced.

The procedure for fingerprinting circuits, i.e., determining the α-values as accurately as possible is the following: (1) Choose a set of circuit inputs. (2) Compute G and $G+$. (3) Perform measurements on a circuit to produce $m+e$. (4) Compute the fingerprint $a' = G+(m+e)$.In this formulation, a' represents the fingerprint that we deciphered from the SVD. It does not necessarily match a, due to the measurement error and also due to gate correlations that hinder gate-level characterization.

Kamel and Sayeed (Kamel & Sayeed, 2008) used Data Glove Technique in signature verification Based on SVD. In the pitch of computer-generated milieus, information glove is an innovative measurement. The first considered to satisfy the strict supplies up-to-date indication apprehension and simulation specialists.

All proposals are relief and luxury of routine such as a minor method issue and several request motorists. Moreover, for truthful real-time simulation, the little cross-correlation and high data amount mark it the best.

The author attempt to modify the application of data glove such as be the sign proof tricky from indicator simulation, he completed routine of the obtainable numerous marks of liberty aimed at apiece finger and hand too.

Singular Value Decomposition is used in the planned system to outcome *r* remarkable vectors recognizing the greatest energy of glove data matrix **A.** This matrix is defined as a major subspace, it is excuse for maximum dissimilarity in the creative information. The dimensionality of the information can be compact.

Having known data glove sign ready its *r*th main subspace, the validity dismisses then be found by scheming the directions among the dissimilar subspaces.

Mention to the lively structures the data glove deliver information such as:

1. Outlines characteristic to an individuals' sign and hand size.
2. Time gone throughout the validation process.
3. Hand route reliant on progressing.

For that reason, the glove as implement for sign appreciation lets verification of people is not just over the biometric features of their signs, but took over the scope of their hands. Figure 4 shows the data glove with the place of the sensors.

Figure 4. Sensor mappings for 5DT data glove 14 ultra

Study a data glove of *m* sensors, apiece marks *n* examples each signature, creating an production data matrix, $\mathbf{A}(m \times n)$. Typically $n >> m$, where *m* means the amount of dignified channels while *n* denotes the number of sizes. Numerous sign processing it has been created requests and regulator schemes that the singular value decomposition of matrix formed from observed data can be used to improve approaches of signal limit approximation and system documentation.

There are two sections for the model of the proposed signature verification technique:

Enrollment Section

- Use data glove to offer the system with ten frank examples of his/her signature.
- Obtainable of the composed ten frank examples choice the orientation signature.
- Excerpt the *r*-principal subspace of the orientation signature and apart from it in the file for equivalent.

Verification Section

- Use data glove to effort the signature of the employer (one sample).
- Compute the *r*-principal subspace of the demanded individuality consuming SVD.
- Competition the main subspace of the demanded individuality to the registered copies in the database over the parallel feature.
- Relate the parallel feature with the choice threshold for ACCEPT or REJECT.

In the way of the *i*th left remarkable course of the matrix **A,** The worried with determination restrained is equivalent to the *i*th singular value squared.

The *r*th principal subspace S_U^r is, among all *r*-dimensional subspaces of R^m, the one that senses a maximal oriented energy. Thus, the orthogonal breakdown of the drive via the singular

value breakdown is official in the intelligence that it lets discovery subspaces of measurement *r* wherever the classification has slight and greatest energy. This breakdown of the ambient space, as straight sum of a space of greatest and slight energy for a assumed vector order, leads to a very stimulating abundant reflection.

Through the launching of the relation among the concerned with energy and SVD, it has been showed that the first *r* left singular courses intelligence the greatest drive of glove data matrix **A**. for that reason, the description for most of the variation in the original data.

This means that with $m \times n$ data matrix that is usually largely over determined with much more samples (columns) than channels (rows): $n >> m$ the singular value decomposition allows to compact most signature characteristics into *r* vectors.

Thus, apiece signature will be recognized over its *r*th main subspace S_U^r, the validity of the strained signature can be got by scheming the viewpoint between its main subspace and the true one.

Gul and Avcibas used Forensic features based on SVD for Cell Phone source identification (Gul & Avcibas, 2007). For the cell-phones equipped with cameras, The documentation of the basis is suitable essential for allowed and safety causes through the always growing obtainability today.

Image source documentation needs a considerate of the physics and procedures of the image creation pipeline.

For nearly all numerical cameras the pipeline is alike, though greatly of the particulars are reserved as branded data of each builder.

Digital camera pipeline contain of a lens scheme, sampler filters, color filter collection, imaging instrument, and a numerical image computer.

There are alterations among numerical cameras and cell-phone cameras.

Whereas their imaging pipelines are like, there are important changes in excellence. The cell-phone cameras effect in minor excellence images owing to numerous details.

They need minor resolve, fixed f/number and minor opening halts. Their flashes are not healthy due to control compels and their analog-to-digital change (ADC) uses 10 bits instead of 12 bits as classically used in conservative numerical cameras.

Documentation method is created on the supposition that the image rows/columns will display the CFA exclamation and device blast typical in the form of comparative direct (in) dependency; as CFA exclamation presents inter pixel associations and device noise is additional in an scene self-governing way.

SVD is a very influential instrument in linear algebra. It rots a matrix $A \in IR$ into the creation of two orthonormal matrices $U \in IR$, $V \in IR$ and a slanting matrix $S \in IR$ as follows:

$$A = USV^T$$

The diagonal rudiments of matrix *S* are nonnegative and organized in lessening instruction, these rudiments produce a vector called remarkable value vector

$$Sv = Diag(S)$$

Singular standards of a matrix direct the soft association between image rows/columns in statuses of linear dependence.

For additional exactly, singular standards tend to develop zero if the image rows and/or columns tend to develop comparatively linearly dependent. Two rows/columns, c1 and c2, of a matrix are named linearly reliant on if they can be defined as $c2 = K \cdot c1$ where *K* is an integer. Consequently, it can be described 'comparative linear dependency' between two rows/columns as the closeness of *K* to an integer.

A worthy perfect of the comparative linear dependency of image rows/columns lead to precisely identify the model of a cell-phone. It can be predictable that the different CFA outburst algorithms of dissimilar mobile phones as well as the noise shaped by the semiconductor de-

vices present revealing properties both on local neighborhoods as well as on image macro blocks, including the image itself. Therefore a common characterization should compromise macro and micro statistics.

In order to obtain micro statistical features, images are first divided into sub-blocks of sizes $w \times w$ (w=3, 4…20). Then, each singular value is regularized with the sum of the remarkable values of the connected sub-block to reduce the dissimilar drive levels of dissimilar images.

For be able to income into explanation the associations within and between the image tablets, the blocks are overlay equivalently to the tablet size.

Macro Statistical Features

Macro statistical features are extracted from the entire image as well as from image macro blocks. The derivation of a singular value vector from an entire image, *Sv*e, is straightforward. For image macro blocks the following procedure is applied to obtain a unique singular value vector:

- Divide the image *A* into four non-overlapping equal size sub-blocks (*A*1, *A*2, *A*3, *A*4).
- Find the mean sub-block *A*s=0.25x (*A*1+*A*2+*A*3+*A*4) and subtract *A*s from the image sub-blocks *B*j = *A*j-*A*s, for *j*=1, 2, 3, 4.
- Calculate singular value vector, *Sv*j, of each *B*j for *j*=1, 2, 3, 4 and determine the average *Sv*, *Sv*a=0.25Σj *Sv*j.
- Normalize *Sv*a with the sum of its elements *Sv*n=(1/K)· *Sv*a where K=Σi *Sv*a(*i*).

CONCLUSION

In this chapter we presented an introduction to SVD and its general applications, SVD thus, proves to be promising domain for security such as watermarking, steganography, encryption, stegoanalysis. Also we introduced some of papers that used SVD in specific computer security. The algorithms presented in this chapter can be improved in several ways.

Due to the arrangement of the singular values in the matrix S (in a descending order), the SVD transformation has the property that the maximal variation among the objects is captured in the first singular value, as $s_1 > s_i$, for $i \geq 2$. Similarly much of the remaining variations are captured in the second dimension, and so on. Thus, a transformed matrix with a much lower dimension can be constructed to represent the original matrix faithfully. This property makes the SVD particularly interesting for our application of high accuracy data hiding.

REFERENCES

Agarwal, R., & Santhanam, M. (2008). Digital Watermarking in the Singular Vector Domain. *International Journal of Image and Graphics*, *8*(3), 351–368. doi:10.1142/S0219467808003131

Andrews, H., & Patterson, C. (1976). Singular value decompositions and digital image processing. *Acoustics, Speech and Signal Processing*. *IEEE Transactions*, *24*(1), 26–53.

Bergman, C., & Davidson, J. (2005). Unitary embedding for data hiding with the SVD. In proceedings of Security, Steganography, and Watermarking of multimedia Contents (pp. 619-630), Bellingham, WA: SPIE.

El Gorfte, Z., Eddeqaqi, N., Bouzid, A., & Roukh, A. (2013). Multi-data embedding in to RGB Image with using SVD method. *International Journal of Computer Science Issues*, *10*(5), 190–195.

Gul, G., & Avcibas, I. (2007). Source Cell Phone Camera Identification Based on Singular Value Decomposition. In proceedings of Signal Processing and Communications Applications Conference (pp. 1-4). Eskisehir, Turket: IEEE.

Kamel, N., & Sayeed, S. (2008). SVD-Based Signature Verification Technique Using Data Glove. *International Journal of Pattern Recognition and Artificial Intelligence*, *22*(3), 431–443. doi:10.1142/S0218001408006387

Liu, R., & Tan, T. (2002). A SVD-Based Watermarking Scheme for Protecting Rightful Ownership. *Multimedia. IEEE Transactions*, *4*(1), 121–128.

Nelson, M., Nahapetian, A., Koushanfar, F., & Potkonjak, M. (2009). SVD-Based Ghost Circuitry Detection. In Information Hiding, Security and Cryptology, (pp. 221-234), Darmstadt, Germany, Springer Berlin Heidelberg.

Rajab, L., Al-Khatib, T., & Al-Haj, A. (2009). Video Watermarking Algorithms Using the SVD Transform. *European Journal of Scientific Research*, *30*(3), 389–401.

Sadek, R. (2012). SVD Based Image Processing Applications: State of The Art, Contributions and Research Challenges. *International Journal of Advanced Computer Science and Applications*, *3*(7), 26–34.

Wang, J., Healy, R., & Timoney, J. (2010). A Novel Audio Watermarking Algorithm Based On Reduced Singular Value Decomposition. In *proceeding of Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference* (pp. 143 – 146), Darmstadt, Germany, IEEE. doi:10.1109/IIHMSP.2010.43

ADDITIONAL READING

Bhat, V., Sengupta, I., & Das, A. (2011). An Audio Watermarking Scheme Using Singular Value Decomposition and Dither-Modulation Quantization. *Multimedia Tools and Applications*, *52*(2-3), 369–383. doi:10.1007/s11042-010-0515-1

El Abbadi, N., Mohamad, A., & Mohammed, M. (2013). Blind Fake Image Detection. *International Journal of Computer Science Issues*, *10*(4), 180–186.

El-Bendary, M., El-Azm, A., El-Fishawy, N., Shawki, F., El-Tokhy, M., El-Samie, F., & Kazemian, H. (2011). SVD Audio Watermarking: A Tool to Enhance the Security of Image Transmission over ZigBee Networks. *Journal of Telecommunications and Information Technology*, (4), 99-107.

Golub, G., & Van Loan, C. (1996). *Matrix Computations* (3rd ed.). London, UK: The Johns Hopkins University Press.

Jagadeesh, B., Kumar, P., & Reddy, P. (2012). Genetic Algorithm approach for Singular Value Decomposition and Quantization based Digital Image Watermarking. *International Journal of Engineering Research and Applications*, *2*(2), 1229–1235.

Kalnins, Y., & Pakalnite, I. (2011). Singular Value Decomposition of Images with the Simple Elements. *Computer Modeling and New Technologies*, *15*(1), 49–54.

Milivojević, Z., & Stevanović, Z. (2013). Analysis on the Robustness SVD-Based Watermarking Algorithms. *International Journal of Computer and Information Technology*, *2*(4), 688–693.

Pomponiu, V., Cavagnino, D., Basso, A., & Vernone, A. (2010). Data Hiding Schemes Based on Singular Value Decomposition. In A. Al-Haj (Ed.), Advanced Techniques in Multimedia Watermarking: Image, Video and Audio Applications (pp. 254-288). Hershey, PA: Information Science Reference. doi:10.4018/978-1-61520-903-3.ch011

Shantikumar, Y., Devi, B., & Singh, Kh. (2013). A Review of Different Techniques on Digital Image Watermarking Scheme. *International Journal of Engine Research*, *2*(3), 193–199.

Singh, N., & Sharma, M. (2010). Singular Value Decomposition Techniquefor Digital Image Watermarking. In *Proceedings of National Conference on Advancements in Wireless and Optical Communication Technologies*, Maharashtra, India.

Swarnalipi, S., Majumder, S., Das, T., & Kumar, S. (2012). Binary Logo Watermarking Based on Multiresolution SVD. *In proceeding of International Conference on Computing and Control Engineering*, (page 61), Chennai, India: Coimbatore institute of information technology.

Wang, J. (2008). *Matrix Decomposition for Data Disclosure Control and Data Mining Applications.* Kentucky, USA: University of Kentucky Doctoral Dissertations.

KEY TERMS AND DEFINITIONS

Encryption: The process of encoding messages (or information) in such a way that third parties cannot read it, but only authorized parties can. Encryption doesn't prevent hacking but it prevents the hacker from reading the data that is encrypted.

Image Processing: Any form of signal processing for which the input is an image, such as a photograph or video frame; the output of image processing may be either an image or a set of characteristics or parameters related to the image.

Information Hiding: The process of embedding information into digital content without causing perceptual degradation.

Security: The field covers all the processes and mechanisms by which computer-based equipment, information and services are protected from unintended or unauthorized access, change or destruction.

Steganalysis: The art and science of detecting messages hidden using steganography; this is analogous to cryptanalysis applied to cryptography.

Steganography: The art and science of encoding hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message. It is a form of security through obscurity.

SVD: The singular value decomposition: is a factorization of a real or complexmatrix, with many useful applications in signal processing and statistics.

Watermarking: The process of hiding digital information in a carrier signal; the hidden information should, but does not need to contain a relation to the carrier signal. Digital watermarks may be used to verify the authenticity or integrity of the carrier signal or to show the identity of its owners. It is prominently used for tracing copyright infringements and for banknote authentication.