UNIT–II

IMAGE

TRANSFORMS

1. UNITARY TRANSFORMS

1.1 One dimensional signals

For a one dimensional sequence $\{f(x), 0 \le x \le N-1\}$ represented as a vector $\underline{f} = \left[f(0)\, f(1) \ldots f(N-1)\right]^T$ of size N, a transformation may be written as

$$\underline{g} = \underline{T} \cdot \underline{f} \Rightarrow g(u) = \sum_{x=0}^{N-1} T(u,x) f(x), 0 \le u \le N-1$$

where $g(u)$ is the transform (or transformation) of $f(x)$, and $T(u,x)$ is the so called **forward transformation kernel**. Similarly, the inverse transform is the relation

$$f(x) = \sum_{u=0}^{N-1} I(x,u) g(u),\ 0 \le x \le N-1$$

or written in a matrix form

$$\underline{f} = \underline{I} \cdot \underline{g} = \underline{T}^{-1} \cdot \underline{g}$$

where $I(x,u)$ is the so called **inverse transformation kernel**.
If

$$\underline{I} = \underline{T}^{-1} = \underline{T}^{*^T}$$

the matrix $\underline{T}$ is called **unitary**, and the transformation is called unitary as well. It can be proven that the columns (or rows) of an $N \times N$ unitary matrix are orthonormal and therefore, form a complete set of **basis vectors** in the $N-$dimensional vector space.
In that case

$$\underline{f} = \underline{T}^{*^T} \cdot \underline{g} \Rightarrow f(x) = \sum_{u=0}^{N-1} T^*(u,x) g(u)$$

The columns of $\underline{T}^{*^T}$, that is, the vectors $\underline{T}_u^* = \left[T^*(u,0)\, T^*(u,1) \ldots T^*(u,N-1)\right]^T$ are called the **basis vectors** of $\underline{T}$.

1.2 Two dimensional signals (images)

As a one dimensional signal can be represented by an orthonormal set of **basis vectors**, an image can also be expanded in terms of a discrete set of **basis arrays** called basis images through a **two dimensional (image) transform**.
For an $N \times N$ image $f(x,y)$ the forward and inverse transforms are given below

$$g(u,v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} T(u,v,x,y) f(x,y)$$

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} I(x,y,u,v) g(u,v)$$

where, again, $T(u,v,x,y)$ and $I(x,y,u,v)$ are called the **forward and inverse transformation kernels**, respectively.
The forward kernel is said to be **separable** if

$$T(u,v,x,y) = T_1(u,x) T_2(v,y)$$

It is said to be **symmetric** if T_1 is functionally equal to T_2 such that

$$T(u,v,x,y) = T_1(u,x)T_1(v,y)$$

The same comments are valid for the inverse kernel.

If the kernel $T(u,v,x,y)$ of an image transform is separable and symmetric, then the transform $g(u,v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} T(u,v,x,y) f(x,y) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} T_1(u,x)T_1(v,y) f(x,y)$ can be written in matrix form as follows

$$\underline{g} = \underline{T}_1 \cdot \underline{f} \cdot \underline{T}_1^T$$

where $\underline{f}$ is the original image of size $N \times N$, and $\underline{T}_1$ is an $N \times N$ transformation matrix with elements $t_{ij} = T_1(i,j)$. If, in addition, $\underline{T}_1$ is a unitary matrix then the transform is called **separable unitary** and the original image is recovered through the relationship

$$\underline{f} = \underline{T}_1^{*T} \cdot \underline{g} \cdot \underline{T}_1^{*}$$

1.3 Fundamental properties of unitary transforms

1.3.1 The property of energy preservation

For a unitary transformation

$$\underline{g} = \underline{T} \cdot \underline{f}$$

and

$$\underline{g}^{*T} = (\underline{T}^{*} \cdot \underline{f}^{*})^T = \underline{f}^{*T} \cdot \underline{T}^{*T}$$

and therefore, by using the relation $\underline{T}^{-1} = \underline{T}^{*T}$ we have that

$$\underline{g}^{*T}\underline{g} = (\underline{f}^{*T} \cdot \underline{T}^{*T})(\underline{T} \cdot \underline{f}) = \underline{f}^{*T} \cdot (\underline{T}^{*T}\underline{T}) \cdot \underline{f} = \underline{f}^{*T}\underline{f} \Rightarrow \|\underline{g}\|^2 = \|\underline{f}\|^2$$

Thus, a unitary transformation preserves the signal energy. This property is called energy preservation property.

This means that every unitary transformation is simply a rotation of the vector $\underline{f}$ in the N-dimensional vector space.

For the 2-D case the energy preservation property is written as

$$\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} |f(x,y)|^2 = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} |g(u,v)|^2$$

1.3.2 The property of energy compaction

Most unitary transforms pack a large fraction of the energy of the image into relatively few of the transform coefficients. This means that relatively few of the transform coefficients have significant values and these are the coefficients that are close to the origin (small index coefficients).

This property is very useful for compression purposes.

2. THE TWO DIMENSIONAL FOURIER TRANSFORM

2.1 Continuous space and continuous frequency

The Fourier transform is extended to a function $f(x,y)$ of two variables. If $f(x,y)$ is continuous and integrable and $F(u,v)$ is integrable, the following Fourier transform pair exists:

$$F(u,v) = \int_{-\infty}^{\infty}\int f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$f(x,y) = \frac{1}{(2\pi)^2}\int_{-\infty}^{\infty}\int F(u,v)e^{j2\pi(ux+vy)}dudv$$

In general $F(u,v)$ is a complex-valued function of two real frequency variables u,v and hence, it can be written as:

$$F(u,v) = R(u,v) + jI(u,v)$$

The amplitude spectrum, phase spectrum and power spectrum, respectively, are defined as follows.

$$|F(u,v)| = \sqrt{R^2(u,v) + I^2(u,v)}$$

$$\phi(u,v) = \tan^{-1}\left[\frac{I(u,v)}{R(u,v)}\right]$$

$$P(u,v) = |F(u,v)|^2 = R^2(u,v) + I^2(u,v)$$

2.2 Discrete space and continuous frequency

For the case of a discrete sequence $f(x,y)$ of infinite duration we can define the 2-D discrete space Fourier transform pair as follows

$$F(u,v) = \sum_{x=-\infty}^{\infty}\sum_{y=-\infty}^{\infty} f(x,y)e^{-j(xu+vy)}$$

$$f(x,y) = \frac{1}{(2\pi)^2}\int_{u=-\pi}^{\pi}\int_{v=-\pi}^{\pi} F(u,v)e^{j(xu+vy)}dudv$$

$F(u,v)$ is again a complex-valued function of two real frequency variables u,v and it is periodic with a period $2\pi \times 2\pi$, that is to say $F(u,v) = F(u+2\pi,v) = F(u,v+2\pi)$

The Fourier transform of $f(x,y)$ is said to converge uniformly when $F(u,v)$ is finite and

$$\lim_{N_1\to\infty}\lim_{N_2\to\infty}\sum_{x=-N_1}^{N_1}\sum_{y=-N_2}^{N_2} f(x,y)e^{-j(xu+vy)} = F(u,v) \text{ for all } u,v.$$

When the Fourier transform of $f(x,y)$ converges uniformly, $F(u,v)$ is an analytic function and is infinitely differentiable with respect to u and v.

2.3 Discrete space and discrete frequency: The two dimensional Discrete Fourier Transform (2-D DFT)

If $f(x,y)$ is an $M \times N$ array, such as that obtained by sampling a continuous function of two dimensions at dimensions M and N on a rectangular grid, then its two dimensional Discrete Fourier transform (DFT) is the array given by

$$F(u,v)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M+vy/N)}$$

$u=0,\ldots,M-1,\ v=0,\ldots,N-1$

and the inverse DFT (IDFT) is

$$f(x,y)=\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(ux/M+vy/N)}$$

When images are sampled in a square array, $M=N$ and

$$F(u,v)=\frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux+vy)/N}$$

$$f(x,y)=\frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(ux+vy)/N}$$

It is straightforward to prove that the two dimensional Discrete Fourier Transform is separable, symmetric and unitary.

2.3.1 Properties of the 2-D DFT

Most of them are straightforward extensions of the properties of the 1-D Fourier Transform. Advise any introductory book on Image Processing.

2.3.2 The importance of the phase in 2-D DFT. Image reconstruction from amplitude or phase only.

The Fourier transform of a sequence is, in general, complex-valued, and the unique representation of a sequence in the Fourier transform domain requires both the phase and the magnitude of the Fourier transform. In various contexts it is often desirable to reconstruct a signal from only partial domain information. Consider a 2-D sequence $f(x,y)$ with Fourier transform $F(u,v)=\Im\{f(x,y)\}$ so that

$$F(u,v)=\Im\{f(x,y\}=|F(u,v)|e^{j\phi_f(u,v)}$$

It has been observed that a straightforward signal synthesis from the Fourier transform phase $\phi_f(u,v)$ alone, often captures most of the intelligibility of the original image $f(x,y)$ (**why?**). A straightforward synthesis from the Fourier transform magnitude $|F(u,v)|$ alone, however, does not generally capture the original signal's intelligibility. The above observation is valid for a large number of signals (or images). To illustrate this, we can synthesise the phase-only signal $f_p(x,y)$ and the magnitude-only signal $f_m(x,y)$ by

$$f_p(x,y)=\Im^{-1}\left[1e^{j\phi_f(u,v)}\right]$$

$$f_m(x,y)=\Im^{-1}\left[|F(u,v)|e^{j0}\right]$$

An experiment which more dramatically illustrates the observation that phase-only signal synthesis captures more of the signal intelligibility than magnitude-only synthesis, can be performed as follows.

Consider two images $f(x,y)$ and $g(x,y)$. From these two images, we synthesise two other images $f_1(x,y)$ and $g_1(x,y)$ by mixing the amplitudes and phases of the original images as follows:

$$f_1(x,y)=\Im^{-1}\left[|G(u,v)|e^{j\phi_f(u,v)}\right]$$

$$g_1(x,y)=\Im^{-1}\left[|F(u,v)|e^{j\phi_g(u,v)}\right]$$

In this experiment $f_1(x,y)$ captures the intelligibility of $f(x,y)$, while $g_1(x,y)$ captures the intelligibility of $g(x,y)$

3. THE DISCRETE COSINE TRANSFORM (DCT)

3.1 One dimensional signals

This is a transform that is similar to the Fourier transform in the sense that the new independent variable represents again frequency. The DCT is defined below.

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

with $a(u)$ a parameter that is defined below.

$$a(u) = \begin{cases} \sqrt{1/N} & u = 0 \\ \sqrt{2/N} & u = 1,\ldots,N-1 \end{cases}$$

The inverse DCT (IDCT) is defined below.

$$f(x) = \sum_{u=0}^{N-1} a(u)C(u)\cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

3.2 Two dimensional signals (images)

For 2-D signals it is defined as

$$C(u,v) = a(u)a(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} a(u)a(v)C(u,v)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

$a(u)$ is defined as above and $u,v = 0,1,\ldots,N-1$

3.3 Properties of the DCT transform

- The DCT is a real transform. This property makes it attractive in comparison to the Fourier transform.
- The DCT has excellent energy compaction properties. For that reason it is widely used in image compression standards (as for example JPEG standards).
- There are fast algorithms to compute the DCT, similar to the FFT for computing the DFT.

4. WALSH TRANSFORM (WT)

4.1 One dimensional signals

This transform is slightly different from the transforms you have met so far. Suppose we have a function $f(x), x = 0,\ldots,N-1$ where $N = 2^n$ and its Walsh transform $W(u)$.

If we use binary representation for the values of the independent variables x and u we need n bits to represent them. Hence, for the binary representation of x and u we can write:
$(x)_{10} = \left(b_{n-1}(x)b_{n-2}(x)\dots b_0(x)\right)_2$, $(u)_{10} = \left(b_{n-1}(u)b_{n-2}(u)\dots b_0(u)\right)_2$
with $b_i(x)$ 0 or 1 for $i = 0,\dots,n-1$.

Example
If $f(x), x = 0,\dots,7,$ (8 samples) then $n = 3$ and for $x = 6$, $6 = (110)_2 \Rightarrow b_2(6) = 1, b_1(6) = 1, b_0(6) = 0$

We define now the 1-D Walsh transform as

$$W(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)\left[\prod_{i=0}^{n-1}(-1)^{b_i(x)b_{n-1-i}(u)}\right] \text{ or}$$

$$W(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)(-1)^{\sum_{i=0}^{n-1} b_i(x)b_{n-1-i}(u)}$$

The array formed by the Walsh kernels is again a symmetric matrix having orthogonal rows and columns. Therefore, the Walsh transform is and its elements are of the form $T(u,x) = \prod_{i=0}^{n-1}(-1)^{b_i(x)b_{n-1-i}(u)}$. You can immediately observe that $T(u,x) = -1$ or 1 depending on the values of $b_i(x)$ and $b_{n-1-i}(u)$. If the Walsh transform is written in a matrix form

$$\underline{W} = \underline{T} \cdot \underline{f}$$

the rows of the matrix $\underline{T}$ which are the vectors $\left[T(u,0)\,T(u,1)\dots T(u,N-1)\right]$ have the form of square waves. As the variable u (which represents the index of the transform) increases, the corresponding square wave's "frequency" increases as well. For example for $u = 0$ we see that $(u)_{10} = \left(b_{n-1}(u)b_{n-2}(u)\dots b_0(u)\right)_2 = (00\dots 0)_2$ and hence, $b_{n-1-i}(u) = 0$, for any i. Thus, $T(0,x) = 1$ and $W(0) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)$. We see that the first element of the Walsh transform in the mean of the original function $f(x)$ (the DC value) as it is the case with the Fourier transform.

The inverse Walsh transform is defined as follows.

$$f(x) = \sum_{u=0}^{N-1} W(u)\left[\prod_{i=0}^{n-1}(-1)^{b_i(x)b_{n-1-i}(u)}\right] \text{ or}$$

$$f(x) = \sum_{u=0}^{N-1} W(u)(-1)^{\sum_{i=0}^{n-1} b_i(x)b_{n-1-i}(u)}$$

4.2 Two dimensional signals

The Walsh transform is defined as follows for two dimensional signals.

$$W(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\left[\prod_{i=0}^{n-1}(-1)^{(b_i(x)b_{n-1-i}(u)+b_i(y)b_{n-1-i}(v))}\right] \text{ or}$$

$$W(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)(-1)^{\sum_{i=0}^{n-1}(b_i(x)b_{n-1-i}(u)+b_i(y)b_{n-1-i}(v))}$$

The inverse Walsh transform is defined as follows for two dimensional signals.

$$f(x,y)=\frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}W(u,v)\left[\prod_{i=0}^{n-1}(-1)^{(b_i(x)b_{n-1-i}(u)+b_i(y)b_{n-1-i}(v))}\right] \text{ or}$$

$$f(x,y)=\frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}W(u,v)(-1)^{\sum_{i=0}^{n-1}(b_i(x)b_{n-1-i}(u)+b_i(y)b_{n-1-i}(v))}$$

4.3 Properties of the Walsh Transform

- Unlike the Fourier transform, which is based on trigonometric terms, the Walsh transform consists of a series expansion of basis functions whose values are only -1 or 1 and they have the form of square waves. These functions can be implemented more efficiently in a digital environment than the exponential basis functions of the Fourier transform.
- The forward and inverse Walsh kernels are identical except for a constant multiplicative factor of $\frac{1}{N}$ for 1-D signals.
- The forward and inverse Walsh kernels are identical for 2-D signals. This is because the array formed by the kernels is a symmetric matrix having orthogonal rows and columns, so its inverse array is the same as the array itself.
- The concept of frequency exists also in Walsh transform basis functions. We can think of frequency as the number of zero crossings or the number of transitions in a basis vector and we call this number **sequency**. The Walsh transform exhibits the property of energy compaction as all the transforms that we are currently studying. (**why**?)
- For the fast computation of the Walsh transform there exists an algorithm called **Fast Walsh Transform (FWT).** This is a straightforward modification of the FFT. Advise any introductory book for your own interest.

5. HADAMARD TRANSFORM (HT)

5.1 Definition

In a similar form as the Walsh transform, the 2-D Hadamard transform is defined as follows.

Forward

$$H(u,v)=\frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\left[\prod_{i=0}^{n-1}(-1)^{(b_i(x)b_i(u)+b_i(y)b_i(v))}\right],\ N=2^n \text{ or}$$

$$H(u,v)=\frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)(-1)^{\sum_{i=0}^{n-1}(b_i(x)b_i(u)+b_i(y)b_i(v))}$$

Inverse

$$f(x,y)=\frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}H(u,v)\left[\prod_{i=0}^{n-1}(-1)^{(b_i(x)b_i(u)+b_i(y)b_i(v))}\right] \text{ etc.}$$

5.2 Properties of the Hadamard Transform

- **Most of the comments made for Walsh transform are valid here.**
- The Hadamard transform differs from the Walsh transform only in the order of basis functions. The order of basis functions of the Hadamard transform **does not** allow the fast computation of it by using a straightforward modification of the FFT. An extended version of the Hadamard transform is the **Ordered Hadamard Transform** for which a fast algorithm called **Fast Hadamard Transform (FHT)** can be applied.
- An important property of Hadamard transform is that, letting H_N represent the matrix of order N, the recursive relationship is given by the expression

$$H_{2N} = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix}$$

6. KARHUNEN-LOEVE (KLT) or HOTELLING TRANSFORM

The Karhunen-Loeve Transform or KLT was originally introduced as a series expansion for continuous random processes by Karhunen and Loeve. For discrete signals Hotelling first studied what was called a method of principal components, which is the discrete equivalent of the KL series expansion. Consequently, the KL transform is also called the Hotelling transform or the method of principal components. **The term KLT is the most widely used**.

Consider a population of random column vectors of the form

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

The **mean vector** of the population is defined as

$$\underline{m}_x = E\{x\}$$

The operator E refers to the expected value of the population, calculated theoretically using the probability density functions (pdf) of the elements x_i.

The covariance matrix of the population is defined as

$$\underline{C}_x = E\{(\underline{x} - \underline{m}_x)(\underline{x} - \underline{m}_x)^T\}$$

The operator E is now calculated theoretically using the probability density functions (pdf) of the elements x_i and the joint probability density functions between the elements x_i and x_j.

Because $\underline{x}$ is n-dimensional, $\underline{C}_x$ and $(\underline{x} - \underline{m}_x)(\underline{x} - \underline{m}_x)^T$ are matrices of order $n \times n$. The element c_{ii} of $\underline{C}_x$ is the variance of x_i, and the element c_{ij} of $\underline{C}_x$ is the covariance between the elements x_i and x_j. If the elements x_i and x_j are uncorrelated, their covariance is zero and, therefore, $c_{ij} = c_{ji} = 0$. The covariance matrix $\underline{C}_x$ can be written as follows.

$$\underline{C}_x = E\{(\underline{x} - \underline{m}_x)(\underline{x} - \underline{m}_x)^T\} = E\{(\underline{x} - \underline{m}_x)(\underline{x}^T - \underline{m}_x^T)\} = E\{\underline{x}\underline{x}^T - \underline{x}\underline{m}_x^T - \underline{m}_x\underline{x}^T + \underline{m}_x\underline{m}_x^T\}$$

It can be easily shown that

$$\underline{x}\underline{m}_x^T = \underline{m}_x \underline{x}^T$$

Therefore,

$$E\{\underline{x}\underline{x}^T - \underline{x}\underline{m}_x^T - \underline{m}_x\underline{x}^T + \underline{m}_x\underline{m}_x^T\} = E\{\underline{x}\underline{x}^T - \underline{m}_x\underline{x}^T - \underline{m}_x\underline{x}^T + \underline{m}_x\underline{m}_x^T\} = E\{\underline{x}\underline{x}^T - 2\underline{m}_x\underline{x}^T + \underline{m}_x\underline{m}_x^T\}$$
$$= E\{\underline{x}\underline{x}^T\} - E\{2\underline{m}_x\underline{x}^T\} + E\{\underline{m}_x\underline{m}_x^T\}$$

Since the vector $\underline{m}_x$ and the matrix $\underline{m}_x\underline{m}_x^T$ contain constant quantities, we can write

$$E\{\underline{x}\underline{x}^T - 2\underline{m}_x\underline{x}^T + \underline{m}_x\underline{m}_x^T\} = E\{\underline{x}\underline{x}^T\} - 2\underline{m}_x E\{\underline{x}^T\} + \underline{m}_x\underline{m}_x^T$$

Knowing that

$$E\{\underline{x}^T\} = \underline{m}_x^T$$

we have

$$\underline{C}_x = E\{\underline{x}\underline{x}^T\} - 2\underline{m}_x E\{\underline{x}^T\} + \underline{m}_x\underline{m}_x^T = E\{\underline{x}\underline{x}^T\} - 2\underline{m}_x\underline{m}_x^T + \underline{m}_x\underline{m}_x^T \Rightarrow$$
$$\underline{C}_x = E\{\underline{x}\underline{x}^T\} - \underline{m}_x\underline{m}_x^T$$

For M vectors from a random population, where M is large enough, the mean vector $\underline{m}_x$ and the covariance matrix $\underline{C}_x$ can be approximately calculated from the available vectors by using the following relationships where all the expected values are approximated by summations

$$\underline{m}_x = \frac{1}{M}\sum_{k=1}^{M}\underline{x}_k$$

$$\underline{C}_x = \frac{1}{M}\sum_{k=1}^{M}\underline{x}_k\underline{x}_k^T - \underline{m}_x\underline{m}_x^T$$

Very easily it can be seen that $\underline{C}_x$ is real and symmetric. Let $\underline{e}_i$ and λ_i, $i = 1,2,\ldots,n$, be a set of orthonormal eigenvectors and corresponding eigenvalues of $\underline{C}_x$, arranged in descending order so that $\lambda_i \ge \lambda_{i+1}$ for $i = 1,2,\ldots,n-1$. Suppose that $\underline{e}_i$ are column vectors.

Let $\underline{A}$ be a matrix whose rows are formed from the eigenvectors of $\underline{C}_x$, ordered so that the first row of $\underline{A}$ is the eigenvector corresponding to the largest eigenvalue, and the last row the eigenvector corresponding to the smallest eigenvalue. Therefore,

$$\underline{A} = \begin{bmatrix} \underline{e}_1^T \\ \underline{e}_2^T \\ \vdots \\ \underline{e}_n^T \end{bmatrix} \text{ and } \underline{A}^T = \begin{bmatrix} \underline{e}_1 & \underline{e}_2 & \ldots & \underline{e}_n \end{bmatrix}$$

Suppose that $\underline{A}$ is a transformation matrix that maps the vectors $\underline{x}$ into vectors $\underline{y}$ by using the following transformation

$$\underline{y} = \underline{A}(\underline{x} - \underline{m}_x)$$

The above transform is called the **Karhunen-Loeve** or **Hotelling** transform. The mean of the $\underline{y}$ vectors resulting from the above transformation is zero, since

$$E\{\underline{y}\} = E\{\underline{A}(\underline{x} - \underline{m}_x)\} = \underline{A}E\{\underline{x} - \underline{m}_x\} = \underline{A}(E\{\underline{x}\} - \underline{m}_x) = \underline{A}(\underline{m}_x - \underline{m}_x) = \underline{0} \Rightarrow$$
$$\underline{m}_y = \underline{0}$$

The covariance matrix of the $\underline{y}$ vectors is

$$\underline{C}_y = E\{(\underline{y} - \underline{m}_y)(\underline{y} - \underline{m}_y)^T\} = E\{\underline{y}\underline{y}^T\}$$

Using the relationships

$$\underline{y} = \underline{A}(\underline{x} - \underline{m}_x)$$
$$\underline{y}^T = [\underline{A}(\underline{x} - \underline{m}_x)]^T = (\underline{x} - \underline{m}_x)^T \underline{A}^T$$

we get

$$\underline{y}\,\underline{y}^T = \underline{A}(\underline{x} - \underline{m}_x)(\underline{x} - \underline{m}_x)^T \underline{A}^T \Rightarrow$$
$$E\{\underline{y}\,\underline{y}^T\} = E\{\underline{A}(\underline{x} - \underline{m}_x)(\underline{x} - \underline{m}_x)^T \underline{A}^T\} = \underline{A}E\{(\underline{x} - \underline{m}_x)(\underline{x} - \underline{m}_x)^T\}\underline{A}^T$$
$$\underline{C}_y = \underline{A}\underline{C}_x\underline{A}^T$$
$$\underline{C}_x\underline{A}^T = \underline{C}_x\begin{bmatrix}\underline{e}_1 & \underline{e}_2 & \dots & \underline{e}_n\end{bmatrix} = \begin{bmatrix}\lambda_1\underline{e}_1 & \lambda_2\underline{e}_2 & \dots & \lambda_n\underline{e}_n\end{bmatrix}$$
$$\underline{C}_y = \underline{A}\underline{C}_x\underline{A}^T = \begin{bmatrix}\underline{e}_1^T \\ \underline{e}_2^T \\ \vdots \\ \underline{e}_n^T\end{bmatrix}\begin{bmatrix}\lambda_1\underline{e}_1 & \lambda_2\underline{e}_2 & \dots & \lambda_n\underline{e}_n\end{bmatrix}$$

Because $\underline{e}_i$ is a set of orthonormal eigenvectors we have that:

$$\underline{e}_i^T\underline{e}_i = 1, i = 1,\dots,n$$
$$\underline{e}_i^T\underline{e}_j = 1, i, j = 1,\dots,n$$

and therefore, $\underline{C}_y$ is a diagonal matrix whose elements along the main diagonal are the eigenvalues of $\underline{C}_x$

$$\underline{C}_y = \begin{bmatrix}\lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n\end{bmatrix}$$

The off-diagonal elements of the covariance matrix of the population of vectors $\underline{y}$ are 0, and therefore, the elements of the $\underline{y}$ vectors are uncorrelated.

Lets try to reconstruct any of the original vectors $\underline{x}$ from its corresponding $\underline{y}$. Because the rows of $\underline{A}$ are orthonormal vectors we have

$$\underline{A}\underline{A}^T = \begin{bmatrix}\underline{e}_1^T \\ \underline{e}_2^T \\ \vdots \\ \underline{e}_n^T\end{bmatrix}\begin{bmatrix}\underline{e}_1 & \underline{e}_2 & \dots & \underline{e}_n\end{bmatrix} = I$$

with I the unity matrix. Therefore, $\underline{A}^{-1} = \underline{A}^T$, and any vector $\underline{x}$ can by recovered from its corresponding vector $\underline{y}$ by using the relation

$$\underline{x} = \underline{A}^T\underline{y} + \underline{m}_x$$

Suppose that instead of using all the eigenvectors of $\underline{C}_x$ we form matrix $\underline{A}_K$ from the K eigenvectors corresponding to the K largest eigenvalues,

$$\underline{A}_K = \begin{bmatrix} \underline{e}_1^T \\ \underline{e}_2^T \\ \vdots \\ \underline{e}_K^T \end{bmatrix}$$

yielding a transformation matrix of order $K \times n$. The $\underline{y}$ vectors would then be K dimensional, and the reconstruction of any of the original vectors would be approximated by the following relationship

$$\hat{\underline{x}} = \underline{A}_K^T \underline{y} + \underline{m}_x$$

The mean square error between the perfect reconstruction $\underline{x}$ and the approximate reconstruction $\hat{\underline{x}}$ is given by the expression

$$e_{ms} = \sum_{j=1}^{n} \lambda_j - \sum_{j=1}^{K} \lambda_j = \sum_{j=K+1}^{n} \lambda_j .$$

By using $\underline{A}_K$ instead of $\underline{A}$ for the KL transform we achieve compression of the available data.

6.2 Properties of the Karhunen-Loeve transform

Despite its favourable theoretical properties, the KLT is not used in practice for the following reasons.

- Its basis functions depend on the covariance matrix of the image, and hence they have to recomputed and transmitted for every image.
- Perfect decorrelation is not possible, since images can rarely be modelled as realisations of ergodic fields.
- There are no fast computational algorithms for its implementation.