

**Karhunen–Loève Transform**  
**(Image Feature Extraction)**

Prateek Narsinghani (0801CS171057)  
Yash Sonkiya (0801CS171094)

## **Contents**

1. Introduction	
1.1 Feature Extraction.....	3
1.2 Karhunen–Loève Transform.....	3
2. Mathematical Formulation.....	4
3. Algorithm.....	6
4. Documentation of API	
4.1 Package Organization.....	7
4.2 Methods.....	7
4.3 Example.....	7
5. Learning Outcome.....	8
A. References.....	9

# **Introduction**

## **1.1. Feature extraction**

Feature extraction involves reducing the number of resources required to describe a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with many variables generally requires a large amount of memory and computation power, also it may cause a classification algorithm to overfit to training samples and generalize poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

## **1.2. Karhunen–Loève Transform**

The KL Transform is also known as the Hotelling transform or the Eigen Vector transform. The KL Transform is based on the statistical properties of the image and has several important properties that make it useful for image processing particularly for image compression. The main purpose of image compression is to store the image in fewer bits as compared to original image, now data from neighboring pixels in an image are highly correlated. More image compression can be achieved by de-correlating this data. The KL transform does the task of de-correlating the data thus facilitating higher degree of compression.

The KL Transform was proposed by Kari Karhunen and Michel Loève

The KL Transform found many applications like: KLT (often called Principal Component Analysis, PCA) is used for both compressing and summarizing information in multi-spectral remote sensing data. The other applications include: Detection of a known continuous signal, Signal detection in white noise and Signal detection in colored noise.

## Mathematical Formulation

The KLT analyzes a set of vectors or images, into basis functions or images where the choice of the basis set depends on the statistics of the image set - depends on image covariance matrix.

- Consider a set of vectors (corresponding for instance to rows of an image)

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \end{bmatrix}$$

First form mean vector of population

$$\mathbf{m}_x = \mathbf{E} [\mathbf{x}]$$

where  $\mathbf{E}[\ ]$  is the 'expectation' or mean operator.

- We define the covariance matrix

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

which equals the outer product of the vector  $\mathbf{x} - \mathbf{m}_x$  with itself. Hence if  $\mathbf{x}$  is a length  $N$  vector,  $\mathbf{C}_x$  is a  $N \times N$  symmetric matrix such that  $\mathbf{C}_x^T = \mathbf{C}_x$ .

- Simplifying we get,

$$\mathbf{C}_x = \mathbf{E} \cdot \mathbf{x}\mathbf{x}^T - \mathbf{E} \cdot \mathbf{m}_x\mathbf{m}_x^T$$

- Find the Eigen values and then the eigen vectors of the covariance matrix  
Eigen value can be found out using the equation

$$|\mathbf{C}_x - \lambda \mathbf{I}| = 0$$

Where,

$\mathbf{C}_x$  = Covariance matrix

$\mathbf{I}$  = Identity Matrix

$\lambda$  = Eigen Value

Eigen Vector can be found out corresponding to each Eigen vector as shown below

$$\mathbf{C}_x \mathbf{v} = \lambda \mathbf{v}$$

(Finding the first Eigen Vector)

- Create the transformation matrix  $\mathbf{T}$ , such that rows of  $\mathbf{T}$  are eigen vectors  
The KL Transformation matrix is formed using the Eigen vectors. Each eigen vector is arranged as a row of the transformation matrix.

- The vector corresponding to the largest Eigen value is placed on the first row and so on.

This KL Transform matrix,  $T$ , is orthogonal i.e.

$$\begin{aligned} T^{-1} &= T' \\ T \cdot T' &= I \end{aligned}$$

- Find the KL Transform

We obtain the KL Transformed image by simply multiplying the Transformation matrix with the centralized image vector  $(x - m_x)$

Therefore,

$$X = T \cdot (x - m_x)$$

This is the formula for KL Transform.

## **Algorithm**

In this section we will give the algorithm to calculate the KL Transform matrix and the transformation.

**Algorithm:** - KL Transform

**Input:** - An image

1. Find the mean vector the covariance matrix  $C_x$  of the given image.
2. Calculate eigen values of the matrix  $C_x$  say  $\lambda_1, \lambda_2 \dots \lambda_n$ .
3. Determine the corresponding eigen values.
4. Sort the eigen values and form a matrix T with vectors as rows, with the vector with highest eigen value at top.
5. Multiply the transformation matrix T with the given image.

# Documentation of API

## 4.1 Package organization

```
from feature_extraction import kl_transform
```

## 4.2 Methods

**kl\_transform(image)**

**Arguments: -**

an image

**Output: -**

**klt**: The KL Transform of image

**eigenVal**: The eigen values

**eigenVec**: The corresponding eigen vectors

## 4.3 Example

```
img = cv2.imread('Lenna_(test_image).png', 0)
klt, eigenVal, eigenVec = kl_transform(img)
cv2_imshow(klt)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

## **Learning Outcome**

- From this we learnt how to extract features of image
- We learnt how to calculate eigen value and eigen vectors of 2d matrix
- KL transform is based on the statistical properties of the image
- It has certain properties that is useful for image processing
- It is used for image compression too



## **References**

1. <http://fy.chalmers.se/~romeo/RRY025/notes/lec17n4.pdf>
2. <http://fourier.eng.hmc.edu/e161/lectures/klt/node3.html>
3. [https://en.wikipedia.org/wiki/Karhunen%E2%80%93Lo%C3%A8ve\\_theorem](https://en.wikipedia.org/wiki/Karhunen%E2%80%93Lo%C3%A8ve_theorem)