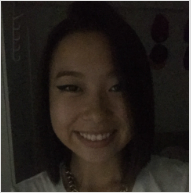
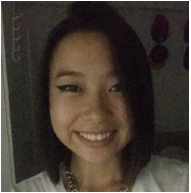
**CHAPTER 3**

**DESIGN AND METHODOLOGY**

**Proposed Solution to the Problems**

**Problem 1:** Sensitivity to illumination cause inaccurate or inability to recognize a person.

*** Training Face Input Image***

***Figure 3.1:*** *Images with different illumination conditions.*

**Objective 1:** Eigenface algorithm that can adapt to various lighting environment settings that will help the recognition to be more accurate and assure the identification of faces.

1. **Existing Algorithm Simulation: Problem 1**

* Get colored image and convert into greyscale image.
* Get image data in form of a ***N x N* matrix** represented as *I*.  
  *The N will be equal to 3 to produce a small scale N X N Matrix.*

*I* = 79 96 83  
 81 103 79   
 103 99 111

*(Lower value indicates darker greyscale pixel)*

* Convert *I* into ***N2x 1*** vector represented as ***Γ***.

***Γ*** = 79  
 96  
 83  
 81  
 103  
 79  
 103  
 99  
 111

**B. Proposed Algorithm Simulation: Problem 1**

* Get colored image and convert into greyscale image.
* Get image data in form of a ***N x N* matrix** represented as *I*.  
  *N will be equal to 3 to produce a small scale sample of N x N matrix.*

*I* = 79 96 83  
 81 103 79   
 103 99 111

*(Lower value indicates darker greyscale pixel)*

* Convert *I* into ***N2x 1*** vector represented as ***Γ***.   
   ***Γ*** = 79  
   96  
   83  
   81  
   103  
   79  
   103  
   99  
   111

* Calculate image data mean and set the state of the illumination.

*Sample mean* ***Γ*** = 79 *If Mean is 0 – 124, Set it to “Dark”* 96 *If Mean is 125 – 190, Set it to “Normal”*   
 83 *If Mean is 191 – 255, Set it to “Bright”*  
 81  
 103 ***Mean* = 92.666 = 93**   
 79  
 103  
 99 *In this case, the* ***Mean*** *is “Dark”.*  
 111

* Calculating the difference of the value of the normal brightness (125) from the *Mean* to adjust image data brightness.

***125 – Mean = Adjust***

***125 – (93) = 32***

*If Mean is “Dark”, add Adjust   
If Mean is “Normal” do nothing*   
*If Mean is “Bright” subtract Adjust*

*In this case, the Adjust will be added.*

***Hence,***

***Γ*** = 79 + 32 ***Γ*** = 11196 + 32 125  
 83 + 32 115  
 81 + 32 113  
 103 + 32 135  79 + 32 111  
 103 + 32 135  
 99 + 32 131 111 + 32 143

***Γ*** = 111 125   
 115   
 113   
 135 ***Mean* = 124.333 = 124**  111   
 135   
 131  143

*(the input image’s overall brightness was increased)*

**Problem 2:** Eigenface Algorithm is susceptible to produces less accurate results compared to the input face image.

**Objective 2:** Eigenface algorithm that will generate a more reliable output and provide the best matches of the inputted image.

1. **Existing Algorithm Simulation: Problem 2**

* Calculate the eigenvalues of the covariance matrix and keep only k largest eigenvalues.
* Compute the eigenvectors of covariance matrix.
* Compute eigenfaces containing highest information of face images
* Compute the projected image

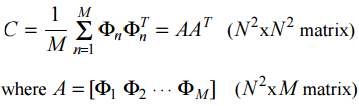
1. **Proposed Algorithm Simulation: Problem 2**

* Identify if the input image’s match exist in the training faces using a common threshold (a set threshold) application.
* Calculate the eigenvalues of the covariance matrix and keep only k largest eigenvalues.
* Compute the eigenvectors of covariance matrix.
* Compute eigenfaces containing highest information of face images.
* Compute the projected image.

**Problem 3:** Inefficiency of PCA’s Dimension Reduction when total number of training faces (M) is larger to the squared of training face dimension (N*2*).  
**Objective 3:** Eigenface algorithm that will maintain and offer a simple yet flexible integration in face recognition systems and sustain a low-level processing that will keep its efficiency.

1. **Existing Algorithm Simulation: Problem 3**

The calculation of the covariance matrix ***C*** is as follows:



Except the matrix ***AAT* (*N2x N2*matrix)** is very large.

*For example:*

*Number of training face (****M)*** = 250   
*Pixel length of the grey scale image* ***N*** = 50

*The calculation for the covariance matrix* ***C*** *would have the result of:*

***C*** *= ((50)2 x (50)2 matrix)*

*C = (2500 x 2500 matrix) <== Too Large*

Following the PCA’s Dimension Reduction, the matrix ***ATA* (*M x M* matrix)** will be considered instead.

*AAT = ATA* (*M x M* matrix)

***C*** *= ((250 x (250) matrix)*

**F. Proposed Algorithm Simulation: Problem 3**

The calculation of the covariance matrix ***C*,** following PCA’s Dimension Reduction, is as follows:

*ATA* (*M x M* matrix)

Except the matrix ***AAT* (*N2 x N2*matrix)** is very large. Following the PCA’s Dimension Reduction, the matrix ***ATA* (*M x M* matrix)** will be considered instead.

However, when the value of ***M*** is comparable to, or even larger than ***N2***, the Dimension Reduction will not be effective.

*For example:*

*Number of training face (****M)*** = 4200   
*Pixel length of the grey scale image* ***N*** = 50

*The calculation for the covariance matrix* ***C*** *would have the result of:*

*C=(4200 x 4200 matrix)<== Too Large*

**In that case, if ***M*** is larger than ***N2***the computation for the covariance matrix ***C*** will be as follows:

*C = ((50)2 x (50)2 matrix)*

*C = (2500 x 2500 matrix)*