Volume 3, No 6; December 2015

Advances in Image And Video Processing

ISSN: 2054-7412

Society for Science and Education - United Kingdom

TABLE OF CONTENTS

EDITORIAL ADVISORY BOARD

DISCLAIMER

All the contributions are published in good faith and intentions to promote and encourage research activities around the globe. The contributions are property of their respective authors/owners and the journal is not responsible for any content that hurts someone's views or feelings etc.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

AIVP ADVANCES IN IMAGE AND VIDEO PROCESSING

ISSN 2054-7412

Using Spectral Decomposition to Detect Dirty Solar Panels and Minimize Impact on Energy Production

[1]Ernesto Zamora Ramos, [2]Suzanna Ho and [3]Evangelos A. Yfantis
Computer Science Department, University of Nevada, Las Vegas, United States;
[1]zamorara@unlv.nevada.edu; [2]hos4@unlv.nevada.edu; [3]yfantis@cs.unlv.edu

ABSTRACT

Dirt and dust deposits on the surface of a solar panel array obstruct the amount of light that can reach the photovoltaic cells, reducing the amount of electricity produced. Solar panels are cleaned when the energy drop has already occurred and is detected. This work presents an algorithm designed to detect dirty solar panels. It is based on the spectral decomposition of the scattered light reflected off the panels' surface by analyzing color images of the surface obtained using digital cameras. It applies the statistical classification method of Mahalanobis distance to separate images where it detects the excess reflected light, classifying them as having a high probability of representing dirty solar panels. It aims to minimize the loss of energy by warning solar plants operators to clean panels before the energy drop becomes significant.

Keywords: Solar Panel, Photovoltaic Cell, Pattern Recognition, Mahalanobis Distance, Classifier.

1 Introduction

Solar power plants are currently growing in number across the globe. They are a source of renewable, clean energy that can be used to significantly reduce the ecological impact and increase the efficiency of production of electric energy.

Solar power plants incorporate large arrays of solar panels. However, today, many individuals have access to solar panels that can be used to produce enough electricity to power a house. Research in the area is abundant right now in pursuit of more efficient ways to collect the solar energy.

Solar panels are collections of interconnected solar cells (also called photovoltaic cells) that absorb the energy of incident light, converting it into an electric current through a phenomenon called "photovoltaic effect."

The photovoltaic (PV) effect is directly related to the photoelectric effect. In summary, the electrons on certain materials can be excited by incident light. Semiconductor materials, such as silicon, are usually used. When an electron in the valence band of a crystal's atom absorbs enough energy from incident photons, it jumps to the conductive band and becomes free, ionizing the source atom with a positive charge. Under the presence of an electric field, the separated electrons and ions are attracted to the opposite charged plates, creating an electromotive force. If a circuit is connected to these plates, an electric current flows. As light continues to excite the material, the ionization is maintained and the electricity continues to flow [1].

Clearly, if more light reaches and gets absorbed by a PV cell, then, more atoms get ionized in the crystal and more electrons become free. As a result, the potential of the electromotive force created by the

DOI: 10.14738/aivp.36.1710
Publication Date: 16th December, 2015
URL: http://dx.doi.org/10.14738/aivp.36.1710

separation of more negative and positive charges increases as well as the electricity flowing through the circuit.

The basic structure of a PV cell is designed to allow the maximum light possible to reach the excitable material, maximize the absorption of photons and minimize reflection. Solar cells rely on a layer of antireflection coating on the front of the cell to reduce reflection of the incident light. On simple cells, light rays enter through the front surface and, if not absorbed, leave through the rear. More sophisticated designs extend the path of light inside the cell to improve absorption through a process called "light trapping" [2].

Sunlight is comprised of ultraviolet, visible, and infrared light. While red light has the longest wavelength in the visible spectrum, infrared light is even longer. Therefore, infrared and red light, both having the larger wavelengths compared to the other components of the incident light, penetrate the glass more readily and produce the most amount of electricity. Meanwhile, blue and violet light suffer the most absorption by the coating of the cell, not the cell itself, and offer little contribution to the production of electric energy in comparison.

As we said earlier, the amount of energy generated by PV cells is directly proportional to the amount of light absorbed. And the more light directly illuminating the cell, more photons reach the material, and more light can be absorbed.

Now, the amount of light striking the solar cells on a solar panel array is dependent on many factors, including the month of the year, day of the month, time of day, weather conditions, and other location–dependent circumstances. Most weather conditions that can limit the amount of light, and thus, the amount of electricity generated, cannot be avoided. Other causes, however, such as light obstruction due to other objects, broken cells and overall cleanliness or dirtiness of the panels, can be dealt with in order to maximize the amount of light reaching the solar cells.

Many solar power plants are established in areas with arid climates due to the low humidity and clear skies year round. Dust and sand storms are common in these climates and the dust gradually settles on the glass surface of solar panels, slowly decreasing the amount of light that reaches the solar cells. The loss of light energy depends on the amount, size, and chemical composition of the dust [3][4]. In terms of time, trees are sparse in arid climates, and during migration in the fall and spring, birds use solar farms as rest areas; therefore, the solar panels become dirty with bird excrement. In general, this is a problem throughout the entire year. Bird droppings are worse than dust, because no light passes through them.

Deposits on a dirty panel reflect, scatter and obstruct the incident light, reducing the amount of photons that can penetrate through and reach the PV cells, consequently decreasing the amount of electricity produced [3].

In this paper, we devise an algorithm specifically to detect dirty solar panels. It is based on the spectral decomposition of the scattered light reflected off the panels' surface by analyzing color images of the surface obtained using digital cameras. It applies statistical classification methods to separate images where it detects the excess reflected light, classifying them as having a high probability of representing dirty solar panels.

It is important to know when a panel is dirty, so that it can be cleaned promptly to minimize the loss of energy. This study aims to warn solar power operators about panels that are becoming dirty before the loss of energy becomes noticeable.

2 Classification Algorithm

The classification algorithm presented here involves sampling, determining the classification vector, and developing, training, and testing the classifier. Our goal is to use videos of solar panel surfaces captured by cameras from a solar panel site and classify the solar panels as clean or dirty. All of the samples we collected are images of solar panels captured by a digital camera [10].

2.1 Observation

By observation, we noted that the surface of solar panels looks dark, due to the limited light reflected off it and the amount of light trapped inside the cells and absorbed by the cells' material. In contrast, a dirty panel's surface looks lighter, because more light is reflected off the surface and scattered by the deposits.

We can see that the corresponding histograms of the tristimulus values (color channels red, green and blue) of images of clean and dirty panels support our hypothesis. See sample in figure 1.

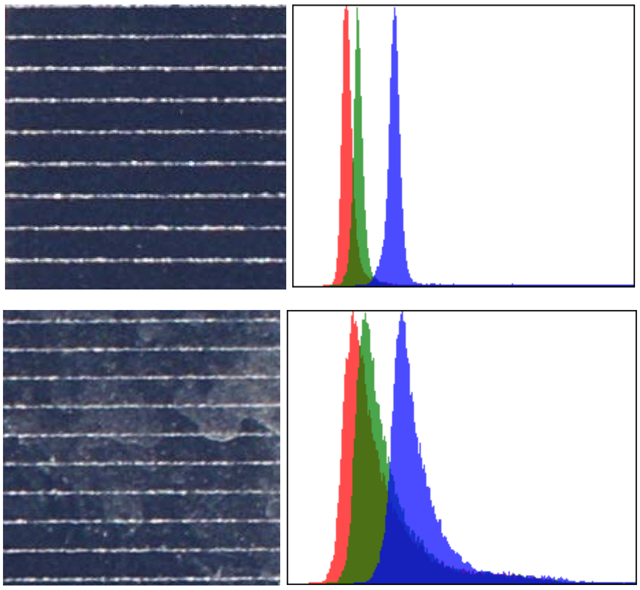


Figure 1. Comparison of a sample for a clean panel with a sample for a dirty panel. Top-left: clean panel; top-right: histogram of the tristimulus values for the clean panel; bottom-left: dirty panel; bottom-right: histogram of the tristimulus values for the dirty panel. Notice that the mean intensity for each color channel is higher for the dirty panel. The variance for each color channel is larger for the dirty panel, indicated by the wider spread of color intensities around the mean intensities.

In general, the mean intensity for each color channel is higher for the dirty panel, indicating a brighter image. This is consistent with our premise that deposits on a dirty panel reflect more light than a clean panel. On the same note, the variance for each color channel is larger for the dirty panel than for the clean panel.

2.2 Sampling

A static digital camera was pointed at a fixed point on a solar panel. The camera capture was utilized to obtain sample data for the classifier development, testing and experimentation phases.

As stated earlier, the intensity on the incident light varies with specific day of the year and time of day. So, for the sampling, we decided to collect data during seven consecutive days, only inside the same timeframe every day, making sure that weather conditions such as wind speed, clouds and temperature remained relatively similar during each collection.

First, we used captured images of a panel when it was clean. Then we artificially applied dirt, dust, or both randomly until we reached the desired power drop to consider the panel dirty, and retrieved the captures. The camera images were divided into samples of dimensions 200×200 pixels. Each obtained image was always cropped in the same way. The goal was to use crops from the same panel, and crops from a combination of panels with similar characteristics. We settled on three groups of training data, where a group contains one clean sample set and one dirty sample set.

1. The First group contains data from the same panel. Each set has 12 samples.
2. The second group builds upon the First group by incorporating data from another panel of similar characteristics, i.e. same PV cell structure. Each set has 20 samples.
3. The third group does not build upon the First and second group. Instead, it incorporates data from two panels of similar characteristics, but with a lighter shade of blue. Each set has 19 samples.

2.3 Classifier development

The goal here is to gauge the similarity of an unknown sample to a known distribution by comparing the set of conditions of the unknown sample to the ideal set of conditions of the known distribution. We can do so by computing the Mahalanobis Distance, which is the relative measure of the data point's distance from a common point [5][6].

Consider a sample set of *K* clean panels, another sample set of *D* dirty panels, and an unknown sample *x* from an arbitrary panel. For this panel, we compute the averages of the tristimulus values (red, green and blue), and the variance–covariance matrix of *x*. The averages constitute a classification vector. If the panel is clean, then its vector belongs to the clean class; otherwise, it belongs to the dirty class. All the clean classification vectors have a grand average, which is another vector, and they have their own variance–covariance matrix. The grand average of the clean classification vectors defines the centroid, or center of gravity, of the clean class. The variance–covariance matrix of the vectors of the clean class defines how close the vectors are to the centroid. Together, the variance–covariance matrix and the centroid make up the parameters of the classifier [9].

The following theorems describe the statistical basis behind the math of our classifier.

2.3.1 Theorem

Let $\overline{x_{cm}}$ be a classification vector from the clean class c of images m = 1, 2, ..., K with centroid $\overline{\overline{x_c}}$ and variance–covariance matrix Σ_c, then the mean of the vector $\overline{x_{cm}} - \overline{\overline{x_c}}$ is zero, and the variance–covariance matrix of $\overline{x_{cm}} - \overline{\overline{x_c}}$ is $\frac{K-1}{K}\Sigma_c$. Furthermore, the Mahalanobis distance from $\overline{\overline{x_c}}$ to $\overline{x_{cm}}$ is

$$d_{cm}^2 = (\overline{x_{cm}} - \overline{\overline{x_c}})^T \left(\frac{K}{K-1}\right) \Sigma_c^{-1} (\overline{x_{cm}} - \overline{\overline{x_c}}) \tag{1}$$

Alternatively, the Mahalanobis distance from $\overline{\overline{x_d}}$ *to* $\overline{x_{dm}}$ *where d is the dirty class is*

$$d_{dm}^2 = (\overline{x_{dm}} - \overline{\overline{x_d}})^T \left(\frac{D}{D-1}\right) \Sigma_d^{-1} (\overline{x_{dm}} - \overline{\overline{x_d}}) \tag{2}$$

Proof: Let $\mu = E(\overline{x_{cm}})$, then:

$$\begin{aligned}
E(\overline{\overline{x}}) &= E\left[\frac{\sum_{m=1}^{K}\overline{x_{cm}}}{K}\right] \\
&= \frac{1}{K}\sum_{m=1}^{K}E[\overline{x_{cm}}] \\
&= \frac{1}{K}K\mu \\
&= \mu
\end{aligned}$$

Thus $E(\overline{x_{cm}} - \overline{\overline{x_c}}) = E(\overline{x_{cm}}) - E(\overline{\overline{x_c}}) = \mu - \mu = 0$, where μ, by definition, is a column vector with three components, namely, the means of the red, green and blue color components.

The variance–covariance matrix Σ of the classification vectors is a 3×3 positive, definite, symmetric matrix and is denoted by:

$$\Sigma = E(\overline{x_{cm}} - \mu)'(\overline{x_{cm}} - \mu)$$

The variance–covariance matrix of the centroid $\overline{\overline{x_c}}$ is:

$$\begin{aligned}
E(\overline{\overline{x_c}} - \mu)'(\overline{\overline{x_c}} - \mu) &= E\left[\frac{\sum_{i=1}^{K} x_{ci}}{K} - \mu\right]'\left[\frac{\sum_{j=1}^{K} x_{cj}}{K} - \mu\right] \\
&= \frac{1}{K^2}E\left[\left(\sum_{i=1}^{K} x_{ci} - \mu\right)'\left(\sum_{j=1}^{K} x_{cj} - \mu\right)\right] \\
&= \frac{1}{K^2}\left\{\sum_{i=1}^{K} E(x_{ci} - \mu)'(x_{ci} - \mu) + \sum_{\substack{i=1 \\ i\neq j}}^{K}\sum_{\substack{j=1 \\ i\neq j}}^{K} E(x_{ci} - \mu)(x_{cj} - \mu)\right\} \\
&= \frac{1}{K^2}K\Sigma + \sum_{\substack{i=1 \\ i\neq j}}^{K}\sum_{\substack{j=1 \\ i\neq j}}^{K} 0 \\
&= \frac{\Sigma}{K}
\end{aligned}$$

The variance–covariance matrix of $\overline{x_{cm}} - \overline{\overline{x_c}}$ is:

$$\begin{aligned}
E(\overline{x_{cm}} - \overline{\overline{x_c}})'(\overline{x_{cm}} - \overline{\overline{x_c}}) &= E[(\overline{x_{cm}} - \mu)(\overline{x_c} - \mu)]'[(\overline{x_{cm}} - \mu)(\overline{x_c} - \mu)] \\
&= E(\overline{x_{cm}} - \mu)'(\overline{x_{cm}} - \mu) - E(\overline{x_{cm}} - \mu)'(\overline{\overline{x_c}} - \mu) + \\
&\qquad - E(\overline{\overline{x_c}} - \mu)'(\overline{x_{cm}} - \mu) + E(\overline{x_c} - \mu)'(\overline{x_c} - \mu) \\
&= \Sigma - \frac{1}{K}\Sigma - \frac{1}{K}\Sigma + \frac{1}{K}\Sigma \\
&= \frac{K-1}{K}\Sigma
\end{aligned}$$

From this result we infer that the Mahalanobis distance from the centroid $\overline{\overline{x_c}}$ to vector $\overline{x_{cm}}$ is:

$$d_{cm}^2 = (\overline{x_{cm}} - \overline{\overline{x_c}})^T \left(\frac{K}{K-1}\right)\Sigma_c^{-1}(\overline{x_{cm}} - \overline{\overline{x_c}}) \quad \square$$

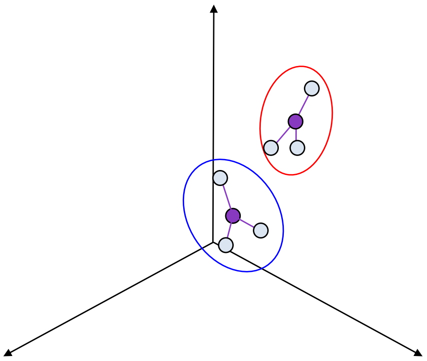


Figure 2. Illustration for theorem 2.3.1. The clean space is blue, and any points within that space are clean classification vectors. The dirty space is red, and any points within that space are dirty classification vectors. The centroids are purple.

Theorem 2.3.1 describes how we can compute the Mahalanobis distance of the *m*-th vector of a class to the centroid of that class, and it is illustrated in figure 2. The clean classification vectors form a space with the centroid $\overline{\overline{x_c}}$, which is a subspace of the 3-D space defined by the (R, G, B) values, and its shape is similar to an ellipsoid, where each axis has a different size. Every classification vector $\overline{x_{cm}}$ of the clean class that does not belong to the intersection between the clean class and the dirty class has a smaller distance from the centroid of the clean space than from the centroid of the dirty space. Similarly, the dirty classification vectors form a space with the centroid $\overline{\overline{x_d}}$, and its subspace has the same characteristics as the clean subspace. Every classification vector $\overline{x_{dm}}$ of the dirty class that does not belong to the intersection of the clean class with the dirty class, has a smaller distance from the centroid of the dirty space than from the centroid of the clean space.

Classification vectors belonging to the intersection of the spaces could have larger distances from the corresponding centroid than from the centroid of the other class. The issue of intersections is addressed later in theorem 2.4.1.

The following theorems describe the statistical basis behind the math of our classifier.

2.3.2 Theorem

Let $\overline{x_{cm}}$*, m = 1, 2, ..., K, be a classification of the clean space with mean vector* μ_c*, variance–covariance matrix* Σ_c*, and centroid* $\overline{\overline{x_c}}$*. Let* $\overline{x}$ *be a new classification vector. If* $\overline{x}$ *belongs to the clean space, then* $E(\overline{x} - \overline{\overline{x_c}}) = 0$*, and the variance–covariance matrix of* $\overline{x}$ *is* $\frac{K-K}{K}\Sigma_c$*. Furthermore, the Mahalanobis distance between* $\overline{x}$ *and* $\overline{\overline{x_c}}$ *is:*

$$d_1^2 = (\overline{x} - \overline{\overline{x_c}})' \left(\frac{K}{K+1}\right)\Sigma_c^{-1}(\overline{x} - \overline{\overline{x_c}}) .$$

Proof: Let $E(\bar{x}) = \mu_c$, then, from theorem 2.3.1, $E(\bar{\bar{x}}_c) = \mu_c$ and $E(\bar{x} - \bar{\bar{x}}_c) = \mu_c - \mu_c = 0$.

The variance–covariance matrix of $\bar{x} - \bar{\bar{x}}_c$ is:

$$\begin{aligned}
E(\bar{x}-\bar{\bar{x}}_c)'(\bar{x}-\bar{\bar{x}}_c) \quad &= E[(\bar{x}-\mu_c)(\overline{x_c}-\mu_c)]'[(\bar{x}-\mu_c)(\overline{x_c}-\mu_c)] \\
&= E(\bar{x}-\mu_c)'(\bar{x}-\mu_c) - E(\bar{x}-\mu_c)'(\bar{\bar{x}}_c-\mu_c) + \\
&\qquad -E(\bar{\bar{x}}_c-\mu_c)'(\bar{x}-\mu_c) + E(\bar{\bar{x}}_c-\mu_c)'(\bar{\bar{x}}_c-\mu_c) \\
&= \Sigma_c - 0 - 0 + \frac{1}{K}\Sigma \\
&= \frac{K+1}{K}\Sigma
\end{aligned}$$

From this result we infer that the Mahalanobis distance from the centroid $\bar{\bar{x}}_c$ to vector $\bar{x}$ is:

$$d_1^2 = (\bar{x}-\bar{\bar{x}}_c)'\left(\frac{K}{K+1}\right)\Sigma_c^{-1}(\bar{x}-\bar{\bar{x}}_c) \quad \square$$

Theorem 2.3.2 describes how we can classify a new, arbitrary vector. The idea has been summarized on figure 3.

2.4 Misclassification Error

In the previous section, we designed our classifier based on the Mahalanobis distance. In summary, we trained the classifier using the sample data that has been classified manually beforehand. The training defines two ellipsoids in space, each representing the respective class of clean panel or dirty panel. When a new, arbitrary sample is to be classified by the trained classifier, we extract the classification vector for the sample and compute the Mahalanobis distance from the new vector to each centroid. Whichever distance is smaller, we say that the new vector, and thus, the new sample belong to that class. This classification decision is justified by theorem 2.4.1.

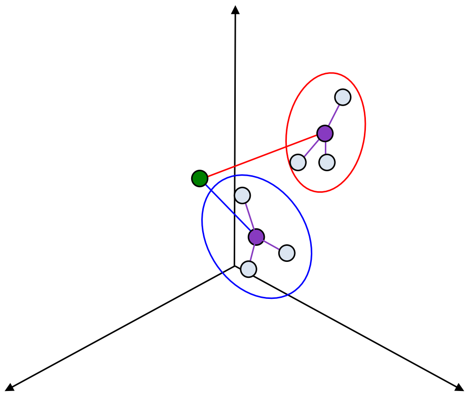


Figure 3. Illustration for theorem 2.3.2. The blue ellipsoid represents the clean class space, and the red ellipsoid represents the dirty class space. The purple points are the centroids of each class. The green point is the unknown vector. It belongs to the clean class since its relative distance to the clean space's centroid is smaller than its relative distance to the dirty space's centroid.

2.4.1 Theorem

Let $\overline{x_{cm}}$, where m = 1, 2, ..., K, be a classification vector of the clean space with mean vector μ_c, variance–covariance matrix Σ_c, and centroid $\bar{\bar{x}}_c$. Let $\overline{x_{dm}}$, where m = 1, 2, ..., D, be a classification *vector of the dirty space with mean vector μ_d, variance–covariance matrix Σ_d which is not significantly different from Σ_c, and centroid $\bar{\bar{x}}_d$. A better estimate of the variance for both classes is $\Sigma = \dfrac{(K-1)\Sigma_c + (D-1)\Sigma_d}{K+D-2}$. Let $\bar{x}$ be a new classification vector. The Mahalanobis distance from $\bar{x}$ to $\bar{\bar{x}}_c$ is:*

$$d_1^2 = (\bar{x}-\bar{\bar{x}}_c)'\left(\frac{K}{K+1}\right)\Sigma(\bar{x}-\bar{\bar{x}}_c)$$

and from $\bar{x}$ to $\bar{\bar{x}}_d$ is

$$d_2^2 = (\bar{x}-\bar{\bar{x}}_c)'\left(\frac{D}{D+1}\right)\Sigma(\bar{x}-\bar{\bar{x}}_c)$$

If $d_1^2 < d_2^2$, then $\bar{x}$ is more likely to be in the clean space, and if $d_1^2 > d_2^2$, then $\bar{x}$ is more likely to be in the dirty space.

Proof: By the central limit theorem, the means for large number of samples are normally distributed [7]. Therefore, if a vector *v* represents the three color channels for a pixel of an image, and $\bar{v}$ is the mean of the channels for all the pixels in the image, then, the probability distribution for the means of a large number of images, $P(\bar{v})$, is normal.

Consider, then, the following functions expressing the probability of $\bar{x}$ belonging to the clean and dirty space respectively:

$$P_c(\bar{x}) = \frac{1}{(2\pi)^{\frac{3}{2}}\left(\frac{K+1}{K}\right)^{\frac{1}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(\bar{x}-\bar{\bar{x}}_c)'\frac{K}{K+1}\Sigma^{-1}(\bar{x}-\bar{\bar{x}}_c)}$$

$$P_d(\bar{x}) = \frac{1}{(2\pi)^{\frac{3}{2}}\left(\frac{D+1}{D}\right)^{\frac{1}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(\bar{x}-\bar{\bar{x}}_d)'\frac{D}{D+1}\Sigma^{-1}(\bar{x}-\bar{\bar{x}}_d)}$$

If we assume that $d_1^2 < d_2^2$, then, $P_c(\bar{x}) > P_d(\bar{x})$ since $\left(\frac{K+1}{K}\right)^{\frac{1}{2}} \approx \left(\frac{D+1}{D}\right)^{\frac{1}{2}} \approx 1$, which implies that $\bar{x}$ is more likely to belong to the clean class. □

Theorem 2.4.1 is the justification of the classification decision. It is summarized in figure 4. But as it can be observed in figure 4, if the spaces overlap, there is the possibility of misclassification.

The power of this classifier is a function of its ability to correctly classify samples. As we saw, there is a chance of incorrectly classifying some new samples; therefore, the higher the probability of the classifier to correctly classify new samples, the more powerful it is. We computed the misclassification error using the Jackknife approach in order to obtain a quantitative measure of the power of our classifier.

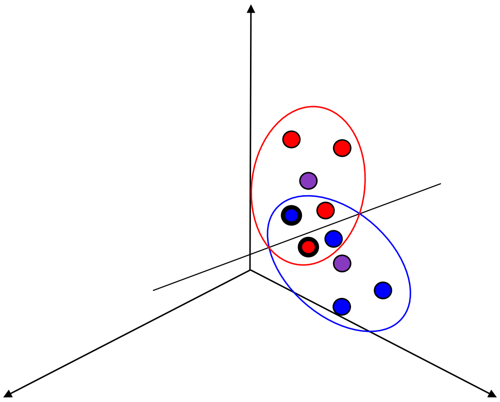


Figure 4. Illustration for theorem 2.4.1. The blue ellipsoid represents the clean class space, and the red ellipsoid represents the dirty class space. The purple points are the centroids of each class. Elements in blue are known to belong to the clean class. Elements in red are known to belong to the dirty class. The black line represents the decision plane created by theorem 2.4.1. The decision plane passes through the adjacency points of the two ellipsoids. All elements (in the case of this representation) above the plane will be classified as clean, while the rest are classified as dirty. All elements will be correctly classified by the theorem, except those with bold outline in the intersection.

2.4.2 Jackknifing

We train the classifier as follows: we use the data from each class to compute the classification vectors for that class, then we use these computed vectors and the parameters obtained by the classifier to define the class. Then we take a vector *v* that already belongs to a class *c*, re-compute *c* without *v*, and then we use our classification algorithm to classify *v*. We repeat this process for each vector from each class, and we estimate the probability of correct classification as the ratio of the number of vectors classified correctly over the total number of classification vectors (see equation 3). This is known as the Jackknifing approach.

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{3}$$

where *TN* (true negative)/*FN* (false negative) are the number of samples correctly/incorrectly classified as clean, and *TP* (true positive)/*FP* (false positive) are the number of samples correctly/incorrectly classified as dirty.

This approach allowed us to find outliers; for example, clean samples with a lighter blue color might be considered dirty because their data behaves similarly to a dirty panel, which means we should not include them with the other clean samples. The more the overlap there is between the set of characteristics that distinguish samples from clean and dirty, the larger the probability of a classification error [8-10]. This was another important concept that was taken care of by the testing method, because it allowed us to filter out samples that barely made the cut; for example, a clean sample could have been correctly classified, but it could have easily been misclassified (i.e. became a false positive) if the dirty and clean samples were slightly different, because sometimes taking out, adding in, or substituting an image upsets the balance. However, we also needed to pay attention to samples that were far from the threshold, known as outliers, because they were too good to be a true negative or positive.

Once the classifier is trained, it continues learning from experience with new samples. If $\overline{x}$ represents the classification vector for a new sample and it is classified correctly to class *c*, then *c*'s space is recalculated, which means a new centroid and variance–covariance matrix is computed.

3 Experimental Results

We used our sample sets as described in section 3.2 to determine the accuracy of our classifier. Table 1 lists the results of the groups of samples computed by using equation 3, as well as the misclassification error, which is the number of false negatives and false positives divided by the total number of classification vectors. It is no surprise that Group 1 has the best accuracy, because it is the strictest. The other two groups have favorable results, considering the data is not as good as it could be.

Table 1. Results of equation 3 from applying Jackknife test on all three groups of training data.

Group	TN	FN	TP	FP	Accuracy (%)	Misclassification Error
1	12	0	12	0	100	0
2	17	3	19	1	90	0.10
3	17	2	19	0	94.4	0.0526

Figures 5, 6, and 7 illustrate the distribution of samples in Group 1, Group 2, and Group 3, respectively. As expected, the graphs do justice to the results listed in table 1, with Group 1 having no overlap due to its 100% accuracy, and groups 2 and 3 having some overlap. The images also reveal unexpected results. Although Group 3 is the least restrictive, its clean and dirty samples are closest to their centroid than the samples of the other two groups. Although Group 1 is the most restrictive, its dirty samples are furthest from their centroid than the other two groups; however, it is the only group where not all pairs of principal components are parallel, specifically the green and blue graph.

Sample images and data collected are available from the authors upon request.

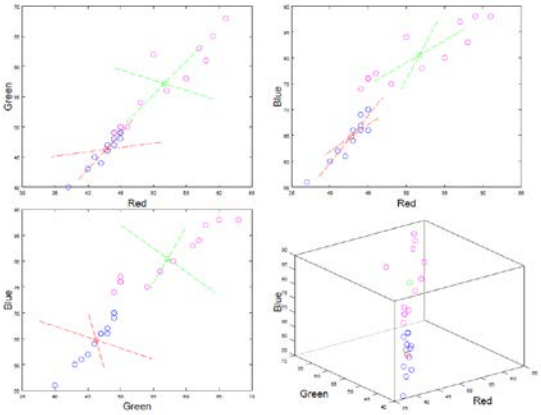


Figure 5. Distribution of Group 1. The blue and magenta circles are clean and dirty samples, respectively. The red and green circle are centroids of the clean and dirty samples, respectively.

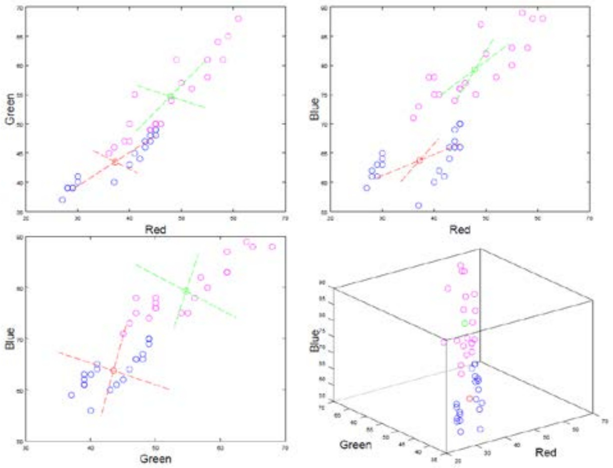


Figure 6. Distribution of Group 2. The blue and magenta circles are clean and dirty samples, respectively. The red and green circle are centroids of the clean and dirty samples, respectively.

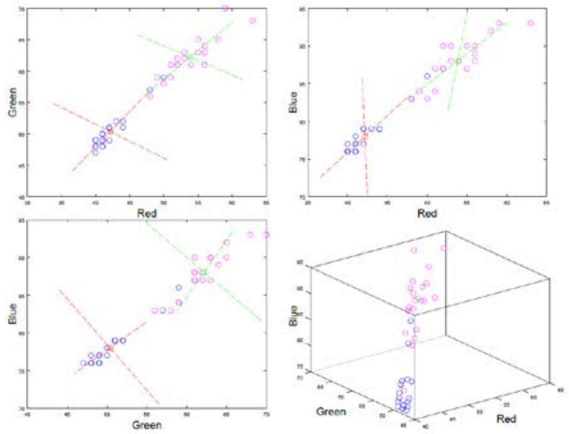


Figure 7. Distribution of Group 3. The blue and magenta circles are clean and dirty samples, respectively. The red and green circle are centroids of the clean and dirty samples, respectively.

4 Conclusions

We presented an algorithm and the supporting mathematical basis to automatically analyze and classify digital images of solar panel surfaces as clean or dirty. The algorithm takes advantage of the observation that photovoltaic cells absorb light, reflecting and scattering very little as a result. Deposits on dirty panels block some light from reaching the photovoltaic cells, reflecting the light back, making the overall image of the surface brighter. Our experimental results show that, if well trained, our algorithm can efficiently and successfully classify dirty solar panels with accuracy above 90%.

Dirty panels are currently detected by inspection or when a power drop occurs. Our algorithm aims to assist solar power operators by warning them about dirty panels before a significant power drop happens, thus minimizing the loss of energy.

5 Acknowledgement

This research is based upon work supported by National Science Foundation under grant number IIA-1301726.

The authors would like to thank Dr. Boehm, professor of Mechanical Engineering at University of Nevada, Las Vegas for allowing access to his solar laboratory and the data obtained from the instruments, and special thanks to Dr. Batista, professor of Civil and Environmental Engineering for her support.

REFERENCES

[1]. K. W. Böer, *Solar Cells*. (2015) Chemistry Explained [Online]. Available: http://www.chemistryexplained.com/Ru-Sp/Solar-Cells.html

[2]. A. J. McEvoy, T. Markvart and L. Castañer, *Principles of Solar Cell Operation.* Practical handbook of photovoltaics: fundamentals and applications, 2nd ed., Academic Press, Elsevier, 2012, pp. 7–30.

[3]. C. P. Ryan, F. Vignola, and D. K. McDaniels, *Solar cell arrays: Degradation due to dirt.* Proceedings of 1989 Annual Conference of The American Solar Energy Society, 1989, pp. 234-237.

[4]. M. K. Mazumder *et al.*, *Solar Panel Obscuration by Dust and its Mitigation in the Martian Atmosphere*. Particles on Surfaces 9: Detection, Adhesion and Removal, 2006, pp. 1-29.

[5]. G. J. McLachlan, *Mahalanobis Distance.* Resonance, Jun. 1999, pp. 20-26.

[6]. E. Zamora Ramos, *Using Image Processing Techniques to Estimate the Air Quality*. McNair Scholars Research Journal, UNLV chapter, 6th ed., 2012, pp 189-194.

[7]. E. W. Weisstein, *Central Limit Theorem.* [Online]. Available: http://mathworld.wolfram.com/ CentralLimitTheorem.html. [Accessed 15 Nov 2015].

[8]. R. Maitra, *Discrimination and Classification – Introduction.* [Online]. Available: http://www.public.iastate.edu/~maitra/stat501/lectures/Classification-I.pdf, 2012. [Accessed 30 May 2014].

[9]. E. A. Yfantis *et al.*, *Pollution Detection in Urban Areas Using the Existing Camera Networks.* International Journal of Multimedia Technology, 2013, Vol. 3, No. 3, pp. 98-102.

[10]. E. A. Yfantis and A. Fayed, *A Camera System for Detecting Dust and Other Deposits on Solar Panels.* Journal Of Advances in Image and Video Processing, 2014, Vol. 2, No. 5, pp. 1-10.

AIVP ADVANCES IN IMAGE AND VIDEO PROCESSING

SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM

ISSN 2054-7412

Application of Simulated Annealing for Color Pattern Recognition to Hybrid Optoelectronic Joint Transform Correlator

Chulung Chen, Yunghsin Hsu, Sihliang Fu, Weichih Liao and Chengsyuan You
Department of Photonics Engineering, Yuan Ze University, Taiwan;
chulung@saturn.yzu.edu.tw

ABSTRACT

For invariant pattern recognition, a method using simulated annealing algorithm is introduced. Near optimal quantized reference functions are designed to be displayed on liquid crystal spatial light modulators. Mach-Zehnder joint transform correlator is adopted as the system for recognition of color targets. From numerical results, the optoelectronic pattern recognition system with simulated annealing algorithm shows a promising capability.

Keywords: Simulated Annealing; Hybrid Optoelectronic Joint Transform Correlator; Color Pattern Recognition.

1 Introduction

VanderLugt correlator (VLC) [1] was proposed for comparing two signals by utilising the Fourier transforming properties of a lens. In 1966, Weaver and Goodman [2] introduced the joint transform correlator (JTC) for pattern recognition application. A few years later, LCD based joint transform correlator (JTC) [3] proposed by Yu and Lu is an attractive tool for pattern recognition. Since then, the JTC configuration has received increased attention because it can be easily implemented. However, the classical JTC suffers from strong zero order term (also called DC term) and broad correlation width. The DC term is the sum of each auto-correlation of the reference image and the target image at the output of correlation plane. The value of the DC term will influence the performance, therefore the removal of the nonzero-order term is of great importance.

Lu et al. [4] utilized phase-shifting technique to design a nonzero- order JTC (N0JTC) and Li et al. [6] used the joint transform power spectrum (JTPS) subtraction strategy to realize the N0JTC. The Mach-Zehnder JTC (MZJTC) [6] can remove the zero-order term in only one step directly without storing the Fourier spectra of both the reference and target images beforehand. Later, Chen et al. [7,8] adopted constraint optimization based on Lagrangian method to yield a sharp correlation peak.

In order to apply reference function in the liquid crystal spatial light modulator, quantized version of the reference template is necessary. On the other hand, the simulated annealing (SA) method [9,10] have been successfully applied to optimization problems. Annealing is a physical process of decreasing temperature slowly in order to reach the global minimum energy states. We will take advantage of this feature for color pattern recognition.

DOI: 10.14738/aivp.36.1742
Publication Date: 28st December, 2015
URL: http://dx.doi.org/10.14738/aivp.36.1742

2 Analysis

The MZJTC structure is shown in Figure. 1. It includes one laser, one spatial filter, one collimated lens (CL), three beam splitters (BS), three polarizing beam splitters (PBS), three Fourier lenses (FL), three reflective liquid spatial light modulators (RLCSLM), three charge coupled device (CCD) cameras, one electronic subtractor (ES) which is used for removing the zero-order term of JPTS, and one computer for controlling the whole system. There are one half wave plate (HWP) and one quarter wave plate (QWP) in front of each RLCSLM. The MZJTC structure is based on the Mach-Zehnder interferometer technique with Stokes relationships. The difference between N0JTC and MZJTC is the MZJTC structure only needs one step to remove the zero-order term of JTPS. The processes are presented as follows.

In the begnining, 3 color component of the test color image are jointly displayed in grascale at the RLCSLM1. In a similar manner, 3 color component of the test color image are displayed in grascale at the RLCSLM2. The target on the RLCSLM1 is illuminated and Fourier optically transformed by FL1. After passing through the PBS3, the irradiation of transmitted and reflected Fourier spectrum is respectively detected by CCD1 and CCD2 in the frequency domain. Then, the difference of joint Fourier power spectrum between CCD1 and CCD2 is displayed at the RLCSLM3. Finally, CCD3 captures its Fouier transform spectrum, which contains the overlapping of each cross-correlation of the reference component and the target component. Meanwhile, the zero-order term is removed. More detailed analysis of MZJTC can be found [10-12].

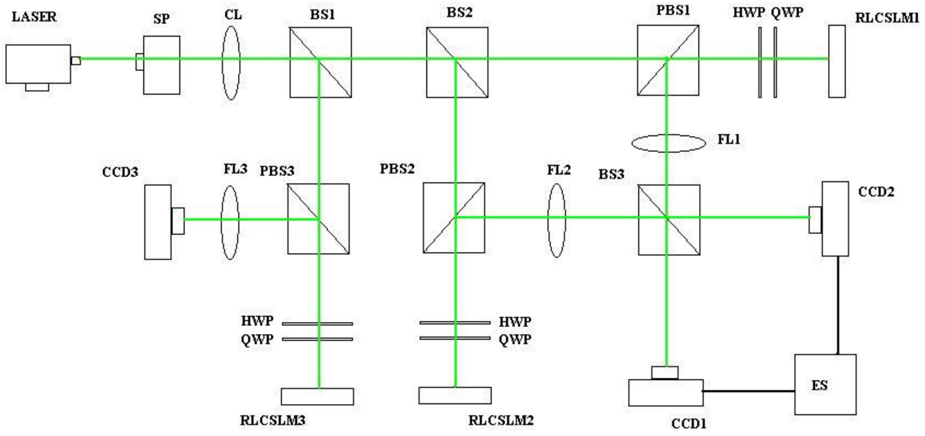


Figure 1. Mach-Zehnder joint transform correlator.

To evaluate the recognition ability, some measurement criteria [13] including correlation peak intensity (CPI) and peak to sidelobe ratio (PSR) are adopted. CPI is the cross-correlation peak intensity at the correlation output plane. PSR is the primary correlation peak energy versus secodary peak energy in the region of interest at the correlation output plane

3 Proposed Algorithm

One colorful butterfly is selected as the basic pattern of the target, whose size is of 64 × 64 × 3 pixels. It is separated into R、G、B channels. For comparison, another butterfly is selected as the nontarget. These two images are shown in Figure 2. For simplicity, We rotate the target in plane from -14° to 14°, and select patterns 2° apart. Totally there are 15 rotationally distorted patterns used as the training set for each color channel. Next, we utilize the training set to obtain a continuous reference function by the constraint optimization technique and then perform the quantization operation on this reference template. The purpose is to obatin a better starting solution within SA algorithm. In our study, the CPE (correlation plane energy) is proposed to construct the energy function.

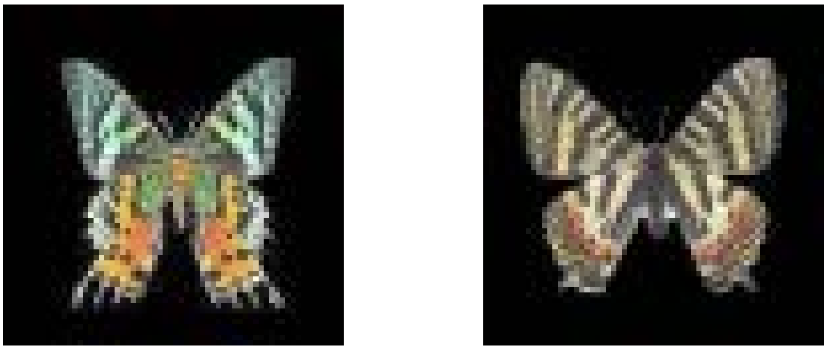


Figure 2: Target (left) and nontarget (right)

A detailed process in SA algorithm for each channel is described as follows:

- Step 1: Yield the initial reference function from constrained optimization technique.
- Step 2: Calculate CPE and CPI for each training image, compute the ratio, and add all the ratios together as the energy function E_{old}. It is expressed as.

$$E_{old} = \sum_{i=1}^{N} \frac{\mathrm{CPE_i}}{\mathrm{CPI_i}} \tag{1}$$

  Here *i* is the index of the training image.

- Step 3: Alter the level number just for one pixel of the reference function *h(x, y)*, and then calculate the new energy function E_{new}.
- Step 4: If the minimum peak value of the new cross-correlation energy function for all training images is not greater than, say, 0.85 times of the minimum peak value of the old cross-correlation energy function, the alteration of the pixel value won’t be accepted and the process returns to the step 3.
- Step 5: Calculate the difference of energy functions, which is ΔE and expressed as

$$\Delta E = E_{new} - E_{old} \tag{2}$$

- Step 6: If $\Delta E \le 0$, accept the level number in the new reference function h(x, y), set E_{new} to be the next time calculated system temperature T, which is the new starting point E_{old}
- Step 7: If not, compute the probability. If it is greater than a random number in the range between 0 and 1, and then accept the alteration of the pixel value.
- Step 8: Check whether all pixels have been operated. If they have, move to the next step. Otherwise go back to step 4.
- Step 9: Record the value of energy function in each cycle. If the normalized standard deviation of energy for the last 10 cycles is smaller than, say, 0.03, and then terminate the computaion and exit the algorithm. Otherwise reduce system temperature by 10%, and go back to step 3.

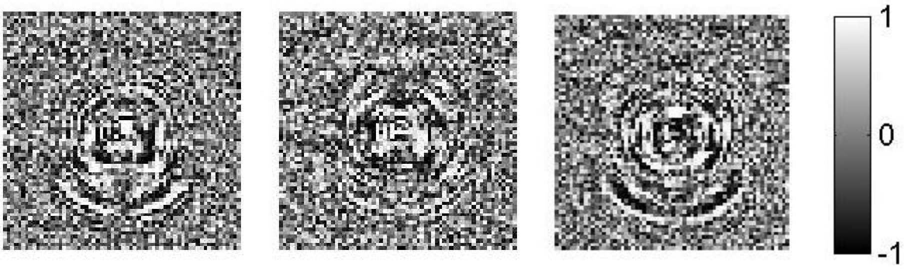


Figure 3: Three reference templates obtained by SA algorithm

4 Result

Figure 3 shows the 3 reference functions in grayscale using 31 levels. Each reference function corresponds to different color component. The dynamic range is also illustrated. It is worthy to notice that the CPI can not be the same for all training targets. Specifically, in our proposed technique, the minimum value of the CPI for these training targets is set as the threshold. Furthermore, the correlation intensity has been normalized to a range between 0 and 1, based on the threshold. Therefore, values above the threshold CPI are set to 1. The CPI curve versus the rotation angle for the target as well as for the nontarget are shown in Figure 4 for the sake of comparison. If the target, for examples, at 11° rotation angle is not in the training set, its CPI value drops below 1. However, the reduction is no more than 20%. The target can still be detected. We can set a threshold value of correlation peak, above which the input can be treated as a target and below which it is a non-target. Therefore, performance is slightly degraded. Figure 5 indicates that the correlator yields PSR values higher than 27 for training subjects. Figure 6 shows the intensity distribution of the correlation output in the region of interest where addition of desired cross correlations of each channel occurs. Both the target and nontarget are 10° rotated. As expected, high correlation peak corresponds to the correct pattern, whereas low correlation profile is observed for the nontarget. We obtain recognition of target and discrimination of nontarget.

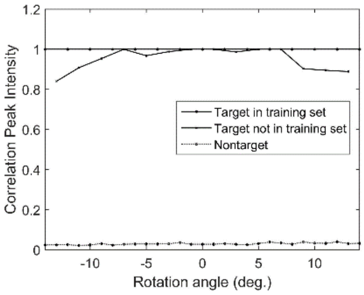


Figure 4: CPI versus rotation angle

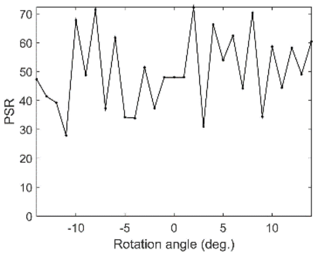


Figure 5: PSR of the target versus rotation angle

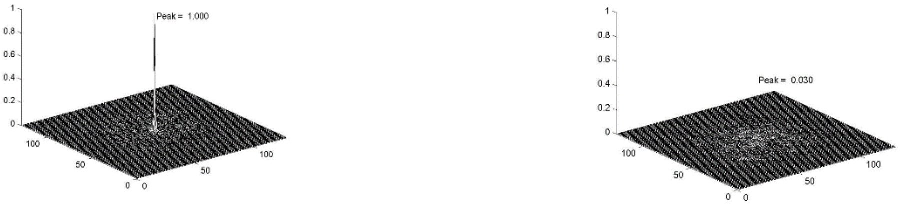


Figure 6: Sample of correlation output for target (left) and nontarget (right) rotated by 10°

5 Conclusion

In this paper, we have proposed a MZJTC with SA for pattern recognition. The SA process will be terminated when the normalized standard deviation of energy function for the last ten iterations is smaller than some value by supposing that convergence is achieved. The performance with SA in the optoelectronic pattern recognition system is promising. The result verifies the feasibility of our proposed method.

REFERENCES

[1]. VanderLugt, A., *Signal detection by complex spatial filtering*. Information Theory, IEEE Transactions on, 1964. 10(2): p. 139-145.

[2]. Weaver, C. S. and J. W. Goodman, *A technique for optically convolving two functions*. Applied Optics, 1966. 5: p. 1248-1249.

[3]. Yu, F. T. S. and X. J. Lu, *A real-time programmable joint-transform correlator*. Optics Communications, 1984. 52: p. 10-16

[4]. Lu, G., et al., *Imentation of a non-zero-order joint-transform correlator by use of phase-shifting techniques*. Applied Optics, 1997. 36: p. 470-483.

[5]. Li, C., S. Yin and F. T. S. Yu, *Nonzero-order joint transform correlator*. Optical Engineering, 1998. 37: p. 58-65.

[6]. Cheng, C. and H. Tu, *Implementation of a nonzero-order joint transform correlator using interferometric technique*. Optical Review, 2002. 9: p. 193-196.

[7]. Wu, C., C. Chen, and J. Fang, *Linearly constrained color pattern recognition with a non-zero order joint transform correlator*. Optics Communications, 2002. 214: p. 65-75.

[8]. Chen C., and J. Fang, *Optimal synthesis of a real-valued template for synthetic aperture radar pattern recognition*. Microwave and Optical Technology Letters, 2002. 32(2): p. 91-95.

[9]. Kirkpatrick, S., et al., *Optimization by simulated annealing*. Science, 1983. 220: p. 671-680,

[10]. Chen, C. and C. Chen, *A Mach-Zehnder joint transform correlator with the simulated annealing algorithm for pattern recognition*. Optics Communications, 2011. 284: p. 3946-3953.

[11]. Fu, S., et al., Application of simulated annealing for color pattern recognition to the optoelectronic correlator with liquid crystal device. The 2012 IAENG International Conference on Imaging Engineering. p.683-688.

[12]. Liu, C., et al., Pattern recognition by Mach-Zehnder joint transform correlator with binary power spectrum," Proceedings SPIE 8559.

[13]. Kumar, B. V. K. V. and L. Hassebrook, *Performance measures for correlation filters*. Applied Optics, 1990. 29: p. 2997-3006.

Society for Science and Education
United Kingdom

AIVP Advances in Image and Video Processing

ISSN 2054-7412

Assessing the performance of Random Partitioning and K-Fold Cross Validation methods of evaluation of a Face Recognition System

Babatunde R. S[1*], Olabiyisi S. O[2], Omidiora E. O[2], Ganiyu R. A[2], Isiaka R. M[1]

[1] *Department of Computer Science*
College of Information and Communication Technology
Kwara State University, Malete. Nigeria

[2] *Department of Computer Science and Engineering*
Faculty of Engineering and Technology
Ladoke Akintola University of Technology, Ogbomoso. Nigeria
ronke.babatunde@kwasu.edu.ng; soolabiyisi@lautech.edu.ng; eoomidiora@lautech.edu.ng;
raganiyu@lautech.edu.ng; abdulrafiu.isiaka@kwasu.edu.ng

ABSTRACT

Face recognition has been an active research area in the pattern recognition and computer vision domains due to its many potential applications in surveillance, credit cards, passport and security. However, the problem of correct method of partitioning the face data into train and test set has always been a challenge to the development of a robust face recognition system. The performance of the System was tested on locally acquired face database when the face database was randomly partitioned and when k-fold Cross Validation partition was used. The face database was captured under the condition of significant variations of rotation, illumination and facial expression. Quantitative evaluation experimental results showed that Random Sampling technique has a higher average recognition rate (96.7%) than Cross Validation partition method (95.3%). However, recognition time in Cross Validation is faster (0.36 secs) than that of Random Sampling (0.38 secs).

Keywords: Pattern Recognition, Cross Validation, k-fold, Random Sampling

1 Background to the Study

Face Recognition has being a broad area of research in the recent years. Its applications are continuously gaining demands due its requirements in person authentication, access control and surveillance systems amongst others (*Thakur et.al, 2010*).Human face cannot be directly used for building automated recognition due to high dimensionality of the face vectors and redundant information contained in the face vectors. The research in face recognition has recently focused on developing a face representation that is capable of capturing the relevant information in a manner which is invariant to facial expression and illumination. If features are inadequately represented, automated face recognition will not be effectively achieved. The classification and subsequent

DOI: 10.14738/aivp.36.1460
Publication Date: 31st,December 2016
URL: http://dx.doi.org/10.14738/aivp.36.1460

recognition time can be reduced by reducing dimension of the image data (*Omidiora, 2006; Omidiora et.al, 2008*). Effective dimensionality reduction encompasses feature extraction and feature selection.

Local Binary Pattern (LBP) feature extraction is a method proposed by (*Ojala, et al*, *2002*). It has been used successfully in a number of applications. The standard way of using LBP-based feature extraction is to evenly distribute patches across an image, so that the whole image is covered. Each patch is of uniform size, and no patches overlap. LBP is then applied to each pixel of a patch resulting in a histogram representing the feature characteristics for that particular patch (*Rose, Reena and Suruliandi, 2011*). A feature vector is created by simply concatenating all of the histograms associated with each patch. These results in transformed features which are suitable for feature selection procedure to select optimal feature subsets (*Babatunde et al, 2014*). The primary purpose of feature selection is to choose a subset of available features, by eliminating features with little or no predictive information and also redundant features that are strongly correlated (*Vieira et al,2010*).

The ACO metaheuristic is characterized as being a distributed, stochastic search method based on the indirect communication of a colony of (artificial) ants, mediated by (artificial) pheromone trails. Ant Colony system involves simple agents (ants) that cooperate with one another to achieve an emergent, unified behaviour for the system as a whole, producing a robust system capable of finding high-quality solutions for problems with a large search space. The pheromone trails in ACO serve as distributed numerical information used by the ants to probabilistically construct solutions to the problem under consideration. The ants modify the pheromone trails during the algorithm's execution to reflect their search experience (*Dorigo and Blum, 2005*). The extraction and selection of the optimal features to represent a face image in a lower dimensional feature space to improve the performance of face recognition systems in terms of time and accuracy is significant.

In this paper, our main objective is to involve the use of the following parameters: training time, recognition time and recognition rate. The values of these parameters were compared on Random Sampling and k-fold Cross Validation. Statistical analysis of the two evaluation methods was carried out. LBP, ACO algorithms and Mahalanobis distance measure was employed for implementing a face recognition system. A locally acquired face database (LAFDAB) which contains photographs of 120 randomly selected individuals was captured with the aid of a 22x HD genx 300 Digital Camera.

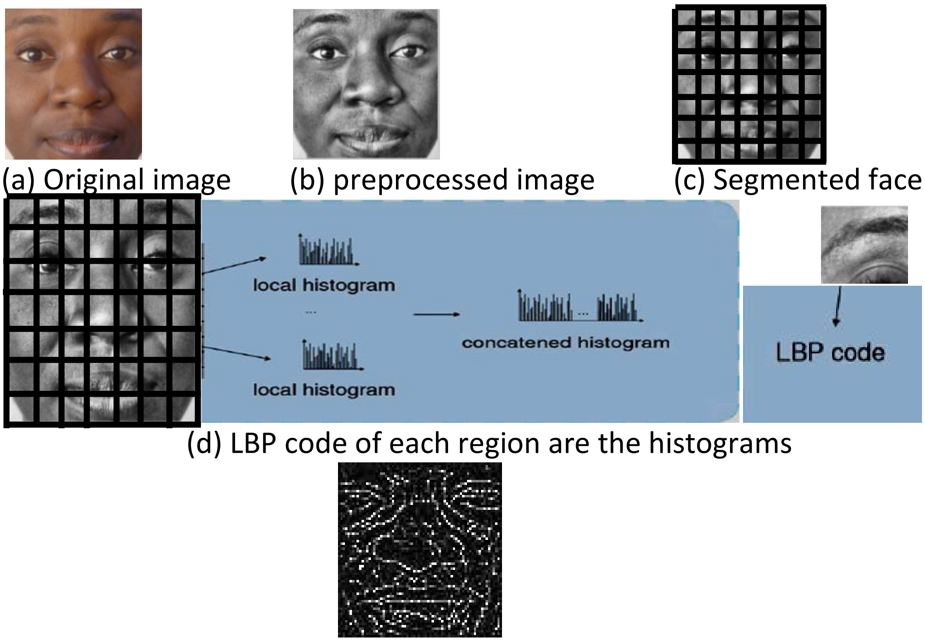
2 Methodology and Procedure of the Face Recognition System

Typically, an RGB face image is normalized and preprocessed as shown in Figure 1a. The face of a subject is initially segmented into a number of uniform, evenly distributed regions that cover the entire image as shown in Figure 1c. LBP code of a pixel captures the structure of local brightness variations around it. The value is computed by sampling circularly around the selected pixel and setting 1-bits in the LBP value for each sample that is brighter than the center pixel using equation 1.

$$LBP(x_c, y_c - \sum_{n=0}^{7} s(i_n, i_c)\, 2^n) \qquad (1)$$

wherei_c corresponds to the grey value of the center pixel (x_c, y_c), in to the grey values of the 8 surrounding pixels. The LBP patterns are obtained by circularly sampling around the center pixel. The effects of circular sampling are that each local neighbourhood is rotated into other pixel location and the sampling point on the circle surrounding the center point are rotated into a different orientation within each neighbourhood (*Ahonen et. al, 2004*).

The original RGB, preprocessed and segmented face images are shown in Figures 3a, b and c. A feature vector describing the textural properties of a given area can be computed by calculating a histogram of the LBP code of each region located inside this area as shown in Figure 3d. The resultant texture feature is shown in Figure 3e.



(a) Original image (b) preprocessed image (c) Segmented face

(d) LBP code of each region are the histograms

(e) Texture of face

Figure 1: sample face image and resultant LBP texture obtained

The matrix shown in Figure 2a was obtained by thresholding the center pixel c, i.e. differences between c and each of its neighbour pixels is calculated. Differences equal or greater than $c$represents value 1 at the pixel position while difference less than c represent 0 at pixel position as shown in the pattern matrix in Figure 2b.

88	93	104
131	98	119
96	89	110

(a) Matrix of a region with pixel values

0	0	1
1		1
0	0	1

Binary 00111001
Decimal 57

(b)Pattern matrix obtained after thresholding the center pixel

Figure 2: The matrix of a sub-region and pattern matrix equivalent

After obtaining the pattern matrix the binary number associated with the pattern matrix is obtained in a clockwise manner as shown by the arrow. The decimal equivalent of the binary value is obtained using binomial weight and this gives the LBP of the region, represented by histogram. The output of LBP was converted to image data matrix to become input into ACO.

The image data matrix will enable the feature selection task of ACO to be formulated by making the output of feature extraction ACO-suitable for selection of optimized feature subset. The image data matrix was created by converting the texture descriptors to double data format in order to set pixels (features) in a double array format so that the matrix of each image can be easily obtained and referenced (*Babatunde et. al, 2015*). The optimal feature subset is obtained using the probabilistic transition rule in equation 2

$$P_{i,j}^{k}(t) = \frac{[\tau_{i,j}(t)]^{\propto}[\eta_{i,j}]^{\beta}}{\sum_{j \in J^k}[\tau_{i,j}(t)]^{\alpha}[\eta_{i,j}]^{\beta}} \text{ , if} i \in J^k \tag{2}$$

where J^k is ant k's unvisited features, η_i is the heuristic desirability of choosing feature *i*, τ_i is the pheromone value at feature i, α determine the importance of pheromone value β determine the importance of heuristic information. J_i^k is the neighborhood of ant *k* when in node *i*. *α* determines the extent to which pheromone information is used as the ants build their solution. *β* determines the extent to which heuristic information is used. The heuristic desirability for this experiment which is the measure of attractiveness of a feature (pixel) based on the local statistics of the image was obtained using the Pearson Product Moment Correlation. The heuristic desirability was obtained by computing the correlation between pairs of pixels. The ants move randomly over the face in a clique to construct a pheromone matrix. The size of the pheromone matrix for this experiment is the resolution of the cropped image (i.e. 70*70), which is arbitrarily chosen. The pheromone trail level and heuristic information are the two most important parameters which determine the success of solution construction in ACO. The process of construction of solution by the ants was carried out by adopting the probabilistic transition rule in equation (2). Once every pixel is visited, a subset of pixels is obtained which represents the optimal set of features (pixels) on the face image salient for face recognition. The resultant optimal feature selected by ACO is shown in Figure 3.

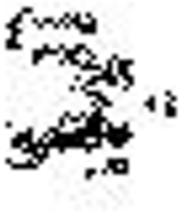


Figure 3: ACO subset image

The most discriminating features in a face pattern, selected by ACO from the face texture were encoded so that comparison between patterns can be made. The feature vectors corresponding to this subset of pixels were used for the recognition process. The Mahalanobis distance between these feature vectors and the test image vector was determined by comparing the covariance between the vectors of the test image and each of the trained images using equation 3.

$$D_i^{Mahalanobis} = \sqrt{(x-\mu i)^T \textstyle\sum_i^{-1}(x-\mu i)} \tag{3}$$

where $\sum_i^{-1}$ represents the inverse of the covariance matrix of class I and μi represents the mean of class I, x is the data point. The Flowchart of the dimensionality reduction process is shown in Figure 4

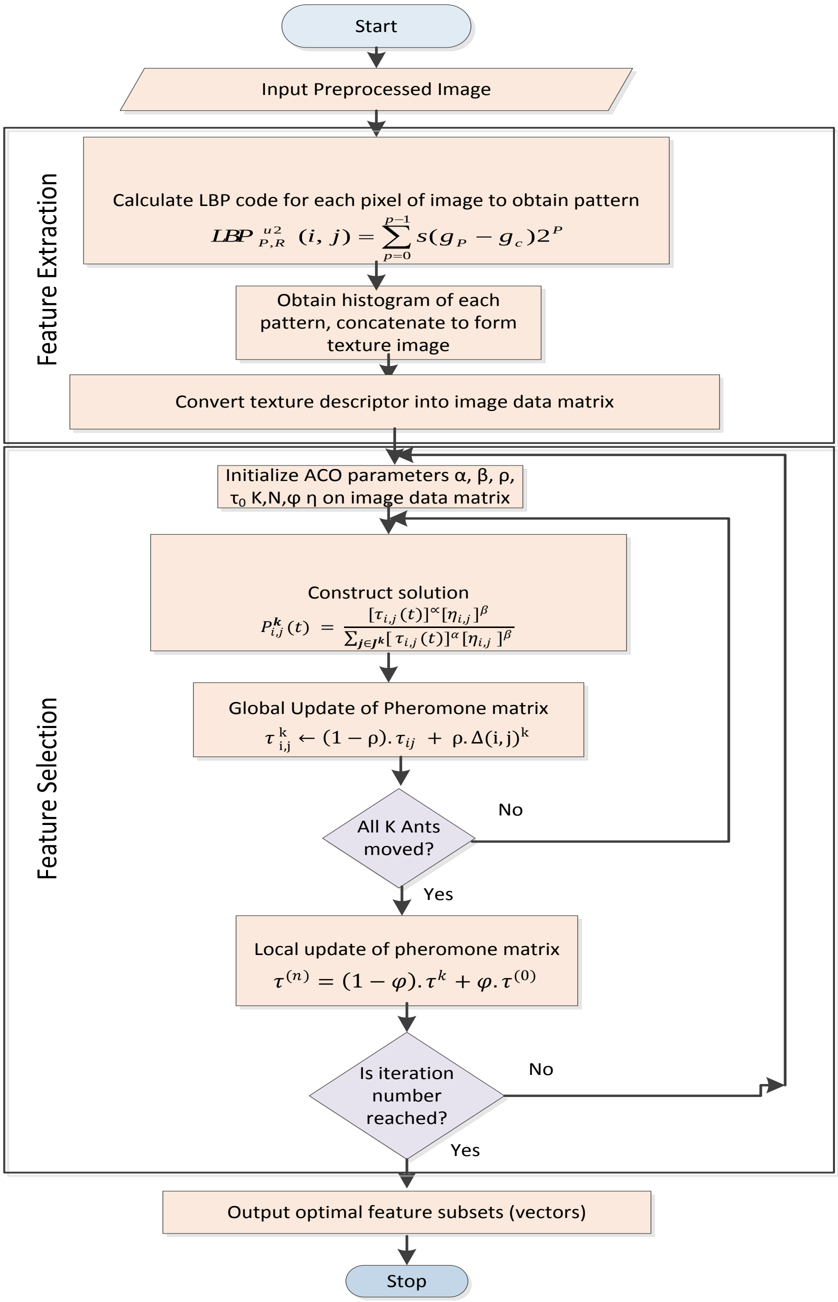

Start
Input Preprocessed Image
Feature Extraction
Calculate LBP code for each pixel of image to obtain pattern
$LBP^{u2}_{P,R}(i,j)=\sum_{p=0}^{P-1}s(g_P-g_c)2^P$
Obtain histogram of each pattern, concatenate to form texture image
Convert texture descriptor into image data matrix
Feature Selection
Initialize ACO parameters α, β, ρ, τ_0 K,N,φ η on image data matrix
Construct solution
$P^k_{i,j}(t)=\frac{[\tau_{i,j}(t)]^{\alpha}[\eta_{i,j}]^{\beta}}{\sum_{j\in J^k}[\tau_{i,j}(t)]^{\alpha}[\eta_{i,j}]^{\beta}}$
Global Update of Pheromone matrix
$\tau^{k}_{i,j}\leftarrow(1-\rho).\tau_{ij}+\rho.\Delta(i,j)^k$
All K Ants moved?
No
Yes
Local update of pheromone matrix
$\tau^{(n)}=(1-\varphi).\tau^k+\varphi.\tau^{(0)}$
Is iteration number reached?
No
Yes
Output optimal feature subsets (vectors)
Stop

Figure 4: Flowchart of Feature Dimensionality Reduction for Face Recognition System

3 Experimental Results and Discussion

The dataset used in our experiment was composed of 720 coloured face images with 6 different images per subject for 120 individuals having a resolution of 1080x1920. The coloured faces were converted to gray scale images, cropped and resized to pixel resolution 70*70. These were carried out in MATLAB 2012R. The images were pre-processed to obtain uniform contrast using contrast limited adaptive histogram equalization technique. This was done because the images were captured locally under uncontrolled and various environmental conditions hence the need to stretch the contrast on the face images to obtain a uniform intensity of brightness. The configuration of the system used is Windows Professional Edition with a 2.4 GHz Intel Core 'i3 processor, 64bit OS and 8 GB of RAM.

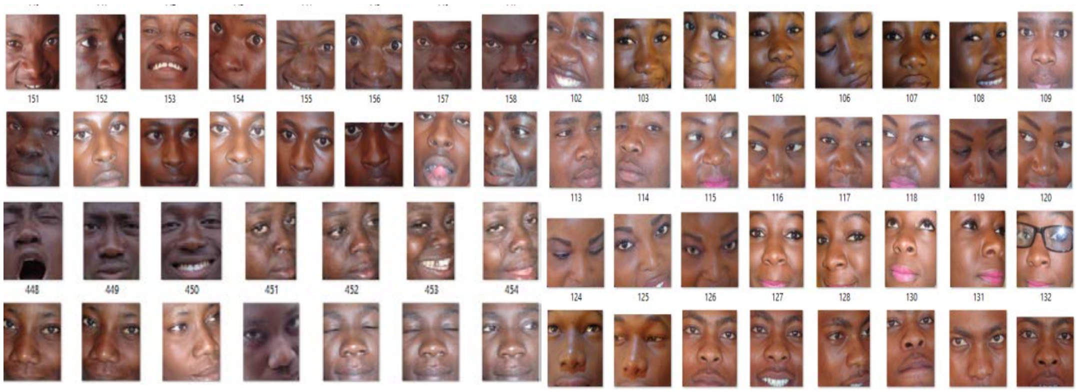


Figure 5: Sample faces from database

3.1 Random partitioning of the database

In this experimental strategy, the whole face images from LAFDAB (720) were randomly partitioned into training and testing sets. For each of the subjects, 4 images were randomly selected as training samples and the remaining 2 images as testing samples. A total of 480 of the 720 faces were used for training and the rest total of 240 facial images was used for testing so as to generate different training and testing sets. The training dataset after pre-processing was subjected to the dimensionality reduction technique to obtain the texture descriptor of the faces as well as optimized feature vectors. The optimal feature vectors were used for training. The result obtained is shown in Table 1

Table 1: Recognition Result of Random sampling of face data

Parameter	Value
Total Training Time (secs)	417.16 secs
Average Training Time (secs)	0.87 secs
Number of images in training set	480
Total Recognition time (secs)	90.06 secs
Average Recognition time (secs)	0.38 secs
Number of images tested	240
Number of images recognized	232
Recognition rate	96.7%

From the Table 1, the average training time obtained from the experiment was 0.87 secs, average recognition time was 0.38 secs and recognition rate was 96.7%.

3.2 Cross Validation Evaluation

We also evaluated the performance of the face recognition system using Cross Validation method. In these experiments, the 720 images in the LAFDAB were divided into 6 folds due to the fact that there are 6 samples of each individual in the database. The images were first divided into 6 folds, with one image of a person in a fold. Hence, each fold consists of 120 images; each one image corresponds to a different person. At each experimental run, 5 folds were used to train and the remaining 1 fold was used for testing. Therefore, the training and testing sets consists of 600 and 120 images respectively in a particular experimental run. The recognition rates for all 6 runs were obtained. The results are shown in Table 2.

Table 2: Result of 6-fold Cross-Validation procedure

Image fold	Training time(secs)	Recognition time(secs)	Total Number of images recognized	Recognition Rate%
Fold1	588.04	43.28	115	95.8
Fold2	578.92	43.01	117	97.5
Fold3	577.38	43.33	114	95.0
Fold4	581.16	42.89	116	96.7
Fold5	583.87	42.91	111	92.5
Fold6	582.41	43.15	113	94.1
Average for 6 folds	589.5	43.09		**95.3**
	0.98secs/image	0.36sec/image		

From Table 2, the average training time for the 6 folds was 589.50secs. Therefore, the average training time for the 600 face images in each training set is 0.98secs per image. Similarly, the average recognition time obtained for the 6 folds was 43.09secs; hence average recognition time per face image could be taken to be 0.36secs. The average recognition rate obtained using 6-fold cross validation method is 95.3%.

3.3 Evaluation of the Results of the two Methods

Inferential Statistical analysis using Paired Sampled t-test was used to analyze the results obtained for Training Time, Recognition Time and Recognition Rate respectively for the two evaluation methods. The Paired Sampled t-test was performed on the null hypothesis (H_0) that there is significant difference between Random Sampling (RS) method and Cross Validation (CV) partition method against the alternative that there is no significant difference (H_1),at 5% level of significance. The hypothesis is defined below;

H_0 : There is significant difference between RS and CV method

H_1 : There is no significant difference between RS and CV method

x = [0.87, 0.38, 96.7] and y = [0.98, 0.36, 95.3]

[h,p,c] = ttest(x,y). The p-value obtained by performing the test was 0.4615. Since p-value is greater than 0.05, we therefore reject the null hypothesis, hence there is no significant difference between the two methods. This signifies that the choice of any of the two methods considered depends on the preference of the researcher involved. However, from the result obtained in the two experiments quantitatively, Random Sampling technique has a higher average recognition rate (96.7%) than Cross Validation method. Additionally, the recognition time obtained using CV partition is smaller than RS by 0.02, indicating a faster recognition time than RS, while the training time in CV is higher than RS.

4 Conclusion

In this research, the performance of Random Partitioning and k-fold Cross Validation methods of evaluation of a Face Recognition System was carried out. This was done to assess the effectiveness of employing any of the evaluation methods. The two methods performed well (and there was no significant difference in the performance of the two methods); hence the choice of any one depends on the preference of the researcher involved. Further research interest hopes to increase the number of folds in the Cross Validation, Kappa Statistics, as well as perform Leave-One-Out Cross Validation partitioning of data.

REFERENCES

[1]. Omidiora E. O,2006: A Prototype of Knowledge-Based System for Black Face Recognition using Principal Component Analysis and Fisher Discriminant Algorithms. Unpublished Ph. D Thesis, Department of Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria.

[2]. Dorigo, M. Stützle, T. (2002): The ant colony optimization metaheuristic: Algorithms applications and advances. In F. Glover and G. Kochenberger, editors, Handbook of Metaheuristics. International series in Operations Research and Management Science. Kluwer Academic Publishers. Vol. 57. pp. 251-285.

[3]. Dorigo, Marco., Blum, Christian. (2005): Ant colony optimization theory. A survey. Theoretical Computer Science Vol. 344 pp. 243 – 278. www.elsevier.com/locate/tcs

[4]. Babatunde R. S, Olabiyisi S.O, Omidiora E.O, Ganiyu R. A. (2014): Feature Dimensionality Reduction using a Dual Level Metaheuristic Algorithm International Journal of Applied Information Systems. Vol. 7(1). pp. 49-52.

[5]. Ojala T., Pietikäinen M., and Mäenpää. T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. (2002): IEEE Transactions on Pattern Analysis and Machine intelligence, Vol 24. pp. 971–987

[6]. Thakur, S. Sing J. K., Basu D. K., Nasipuri, M., Kundu M. (2010): Face Recognition using Principal Component Analysis and RBF Neural Networks. IJSSST, Vol. 10(5).

[7]. Omidiora E.O., Fakolujo O.A., Ayeni R.O., Olabiyisi S.O., and Arulogun O.T. (2008): Quantitative Evaluation of Principal Component Analysis and Fisher Discriminant Analysis Techniques in Face images. Journal of Computer and its Applications. Vol.15 (1). pp. 22-37.

[8]. Rose R. Reena and Suruliandi A. (2011): Improving Performance of Texture Based Face Recognition Systems by Segmenting Face Region. International Journal of Network Security, Vol. 02, No. 03.Pp 23-27.

[9]. Babatunde R. S, Olabiyisi S. O, Omidiora E. O, Ganiyu R. A. (2015): Local Binary Pattern And Ant Colony Optimization Based Feature Dimensionality Reduction Technique For Face Recognition System. British Journal of Computer Science and Mathematics(Article in Press)

[10]. Kashef S., Nezamabadi-pour H. (2014). An advanced ACO algorithm for feature subset selection. Neurocomputing. pp. 1-9. http://dx.doi.org/10.1016/j.neucom.2014.06.067i

[11]. Kavita, Chawla H.S. and Saini J.S. (2011): Parametric comparison of Ant colony optimization for edge detection problem. International Journal of Computational Engineering & Management, 13:54-58

[12]. SodhiKuldeep Singh and Lal Madan. (2013): Comparative Analysis of PCA-based Face Recognition System using different Distance Classifiers. International Journal of Application on Innovation in Engineering and Management. Vol 2(7):341-348.