

# Fly Wing Biometrics Using Modified Local Binary Pattern, SVMs and Random Forest

Foysal Ahmad, Kaushik Roy, Brian O'Connor, Joseph Shelton, Gerry Dozier, and Ian Dworkin

**Abstract**—This paper presents an efficient approach for classification of the gender of a common fruit fly, *Drosophila melanogaster*, based on their wing's texture. The novelty of this research effort is that a Modified Local Binary Pattern (MLBP), which combines both the sign and magnitude features for the improvement of fly wing's texture classification performance, is applied. The extracted features are then used to classify the gender of the fruit fly by using the Support Vector Machines (SVMs) and Random Forest (RF). We validate the performance of the proposed scheme on two fly wing datasets. The highest accuracy achieved by the proposed approach is 94%. In this paper, we limit our approach to gender classification; however, this effort can be extended to explore important characteristics of a fly using wing's texture analysis.

**Index Terms**—Biometrics, fly wing, modified local binary pattern, random forest, and support vector machine.

## I. INTRODUCTION

Biometrics refers to the science of recognizing humans by utilizing the physical (e.g., fingerprint, face, iris) and/or behavioral (e.g., gait, signature) traits of an individual. Although biometrics processes are generally applied on human subjects, in this research effort, we examine the wing's texture of a common fruit fly, *Drosophila melanogaster*, to classify its gender and this classification process can be referred as fly wing biometrics. Fig. 1 shows samples of female and male wings of *Drosophila*. Research during the past decades has demonstrated that humans and flies are similar on the cellular and molecular level [1]-[3]. Human being is different from flies; however, some of the proteins and genes that are required to shape a fly wing are similar to those in humans. Moreover, research works on flies as a model system are explored to better understand the human diseases [1]-[3]. Although we analyze the texture of a wing to ramify *Drosophila* gender only, our research on fly wing texture can help biologist and entomologist to explore the genetic evaluation and development of different insects and their effects on nature.

Texture analysis aiming to interpret and understand real-world visual patterns is an active and challenging research field. Local binary pattern (LBP) proposed by Ojala *et al.* [4] has been considered as an effective descriptor for

texture classification. It has many attractive properties such as rotation invariance, low computational burden, and robustness against monotonic gray level transformation. The LBP-based feature extractor has proven to be highly distinctive and its key advantages, including the robustness against illumination and pose variations, and the computational efficiency, make it suitable for high level image analysis tasks. Despite the great success of LBP in computer vision and pattern recognition, the various extensions and modifications of this texture extraction approach have been proposed [5]-[7].

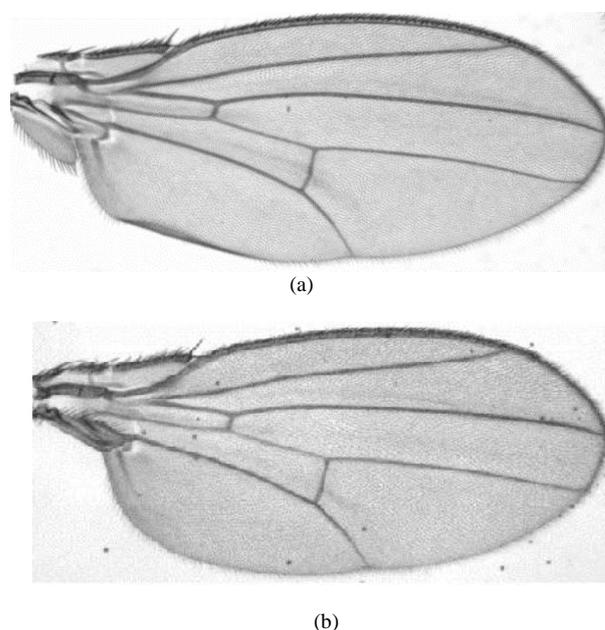


Fig. 1. Wing of a (a) female and (b) male *Drosophila*.

The first approach on classification of gender of *Drosophila* using their wing's texture was proposed in [8]. Shelton *et al.* [8] applied the genetic and evolutionary feature extraction with machine learning (GEFE<sub>ML</sub>) in an effort to elicit the wing's textural features. GEFE<sub>ML</sub> technique evolved the LBP based feature extractors and consisted of a set of patches in various positions. These patches were overlapped with each other and concentrated on smaller and discriminating areas of an image while standard LBP considered the entire image for texture analysis. To the best of our knowledge, this is the second work on fly wing biometrics. In this research effort, instead of using the traditional LBP, we use modified LBP (MLBP) [5] in an effort to increase the gender classification accuracy. The MLBP technique fuses both the sign and magnitude feature components to improve the feature extraction performance; whereas the traditional LBP utilizes only the sign information. Even though the sign component of LBP operator preserves

Manuscript received October 29, 2013; revised January 3, 2014. This research work was supported by Science and Technology Center: Bio/Computation Evolution in Action Consortium (BEACON).

Foysal Ahmad, Kaushik Roy, Brian O'Connor, Joseph Shelton, and Gerry Dozier are with the North Carolina A&T State University, Greensboro, NC 27411 USA (e-mail: fahmad@aggies.ncat.edu; kroy@ncat.edu; bpoconno@aggies.ncat.edu; jashelt1@aggies.ncat.edu; gvdozier@ncat.edu).

Ian Dworkin is with Michigan State University, East Lansing, Michigan 48824 (e-mail: idworkin@msu.edu).

most of the information of local differences, the magnitude component provides additional discerning features that can improve the overall classification accuracy. The extracted MLBP features are then trained and validated by Support Vector Machines (SVMs) and Random Forest (RM) techniques to identify the gender of *Drosophila* by exploring the texture of their wings.

The rest of this paper is organized as follows. Section II briefly describes the modified LBP. Sections III and IV present feature classification techniques using SVMs and RF. Section V reports the extensive experimental results and Section VI provides our conclusions.

## II. FEATURE EXTRACTION USING MODIFIED LBP

The LBP method was first proposed by Ojala *et al.* [4] to encode the pixel-wise information in images. Images are probed locally by sampling grayscale values at a central point  $g_c$  and  $P$  points at  $g_1, g_2, \dots, g_{P-1}$  spaced equidistantly around a circle of radius  $R$ .

$$LBP_{P,R} = \sum_{p=1}^{P-1} r(g_p - g_c)2^p, \quad r = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (1)$$

Given an  $I^*J$  image, let  $LBP_{P,R}(i, j)$  be the identified LBP pattern of each pixel  $(i, j)$ , then the whole texture image is represented by a histogram vector  $Hist$  of length  $K$

$$Hist(k) = \sum_{i=0}^I \sum_{j=0}^J f(LBP_{P,R}(i, j), k), \quad k \in [0, K] \quad (2)$$

$$f(x, y) = \begin{cases} 1, & x = y \\ 0, & x \neq y \end{cases}$$

where  $K = 2^P$  is the number of all the LBP codes. Feature histogram  $Hist$  contains properties of gray-scale invariance, low complexity, few parameters, and satisfactory discriminating power. With the  $P$  neighbors around a pixel, the basic LBP operator produces  $2^P$  distinct features and it becomes a complex problem to analyze feature vector due to overwhelming dimensionality of  $Hist$  with large  $P$ . To reduce the number of binary patterns, Ojala *et al.* [4] suggested an improvement with uniform patterns and merged all non-uniform patterns into one pattern. In uniform pattern, there is zero or only 2 transitions of bits in the circular binary presentation. In the uniform patterns of  $P$  neighbors, the feature vectors are reduced to  $P \times (P-1) + 3$  instead of  $2^P$ . These uniform patterns provide a vast majority, sometime over 90 percent, of a  $3 \times 3$  texture pattern of the entire fly wing image. These patterns, moreover, correspond to the primitive micro features, such as edges, corners, and spots [4].

For MLBP, we calculate the difference of a pixel  $g_c$  with its circularly and evenly spaced  $P$  neighbors  $g_p, P=0, 1 \dots P-1$  as  $D_p = g_p - g_c$ . The difference vector  $D_p$  contains both the sign and magnitude information of the local differences:

$$D_p = S_p \times M_p \text{ and } \begin{cases} S_p = \text{sign}(D_p) \\ M_p = |D_p| \end{cases} \quad (3)$$

where  $S_p = \begin{cases} 1, & D_p \geq 0 \\ -1, & D_p < 0 \end{cases}$  and  $M_p$  are the sign and magnitude components of  $D_p$ , respectively. Due to sign and magnitude

information, the  $D_p$  is more robust against the illumination changes. Fig. 2 shows an example of MLBP feature extraction technique. Fig. 2(a) is the original  $3 \times 3$  structure with central pixel being 34. The difference vector, shown in Fig. 2(b), is  $[-13, -21, 19, -28, -6, -10, 31, 12]$ . After the sign and magnitude components are extracted, the sign (see Fig. 2(c)) and magnitude vectors [see Fig. 2(d)] are represented as  $[-1, -1, 1, -1, -1, -1, 1, 1]$  and  $[13, 21, 19, 28, 6, 10, 31, 12]$ , respectively. The original LBP exploits only the sign vector to code the local pattern as an 8-bit string "00100011" (-1 is coded as 0 for LBP). In the following sections, we provide an overview of SVMs and RF approaches.

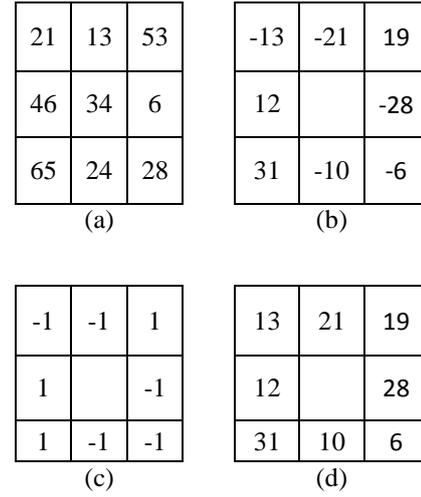


Fig. 2. Feature extraction process using MLBP (a)  $3 \times 3$  sample block, (b) local difference, (c) sign components, and (d) magnitude components.

## III. SUPPORT VECTOR MACHINES (SVMs)

The Support Vector Machine (SVM) is a state-of-the-art classification method introduced in 1992 by Cortes and Vapnik [10]. The SVM classifier is widely used in bioinformatics due to its high accuracy and ability to deal with high-dimensional data such as gene expression, and flexibility in modeling diverse source of data [11]. SVM performs classification tasks by constructing hyperplanes in a multidimensional space that separates cases of different class labels. SVM supports both the regression and classification tasks and can handle multiple continuous and categorical variables. To construct an optimal hyperplane, SVM employs an iterative training algorithm, which is used to minimize an error function. Given a training set of instance-label pairs  $(x_i, y_i), i = 1 \dots N$  where  $x_i \in R^n$  and  $y_i \in \{1, -1\}^N$ , the SVMs require the solution of the following optimization problem:

$$\frac{1}{2} w^T w + C \sum_{i=1}^N \xi_i \quad (4)$$

Subject to the constraints:

$$y_i(w^T \phi(x) + b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0, i = 1, \dots, N$$

where  $C$  is the penalty parameter,  $w$  is the vector of coefficients,  $b$  is a constant, and  $\xi_i$  represents parameters for handling non-separable data (inputs). The index  $i$  labels the

$N$  training cases,  $y \in \pm 1$  represents the corresponding class labels, and  $x_i$  represents the independent variables. Training vectors,  $x_i$  are mapped into a higher dimensional space by the function  $\phi$ . The larger the  $C$ , more the error is penalized. Thus,  $C$  is chosen with care to avoid over fitting. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

$$K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j) \quad (5)$$

Equation (5), the kernel function, represents a dot product of input data points mapped into the higher dimensional feature space by the transformation,  $\phi$ . The four basic kernel functions used in SVM are:

- Linear:

$$K(x_i, x_j) = x_i^T x_j$$

- Polynomial:

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$$

- Radial Basis Function (RBF):

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$$

- Sigmoid:

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$$

where  $\gamma$ ,  $r$  and  $d$  are the kernel parameters.

#### IV. RANDOM FOREST (RF)

Random Forest is an ensemble of unpruned classification or regression trees created by using bootstrap samples of the training data and random feature selection in tree induction. Prediction is made by aggregating (majority vote or averaging) the predictions of the ensemble [12]. Boosting and bootstrapping or bagging are two well known methods for ensemble learning of classification tree. The sample subsets creation for a tree is dependent on previous classification results and addition weights are given to the samples that are incorrectly predicted previously. In bagging method, the trees are developed randomly using a bootstrap sample of the data set in the bagging method and these tree construction do not depend on earlier trees [13], [14]. In RFs, an additional layer of randomness is included to the bagging process. RF constructs each tree using a different bootstrap sample of the data and change the method of the classification or regression trees creation. When developing individual trees, an arbitrary subset of attributes is drawn from the best attributes for the split is selected. In standard trees, each node is created using the best split among all the variables. The classification is based on the majority vote from individually developed tree classifiers in the forest. The specific size of the subset is a parameter of the forest. RFs are fast and easy to implement, produce highly accurate predictions and can handle a very large number of input variables without overfitting [13].

In Ref. [12], the largest possible trees are grown without pruning. Since the unpruned trees are low-bias and high variance models, averaging over an ensemble of trees reduce variance while keeping low bias. The root node of each tree contains a different bootstrap sample which is randomly selected from the original training data. The leaves of a tree provide the feature elements of the same class label. The class label of a new data is predicted based on the leaf in which that data lands. The RF algorithm can be summarized as follows [13]:

- 1) Extract  $n_t$  bootstrap samples from the original data.
- 2) For each  $n_t$ , develop an unpruned classification or regression tree. At each node, randomly sample  $m_t$  of the predictors and select the best split among those variables.
- 3) Predict new data by aggregating the predictions of the  $n_t$ . The majority votes and average are used for classification and regression, respectively.

In RF, an estimate of the error rate can be measured, based on the training data, according to following [13]:

- 1) At each iteration of bootstrap process, predict the data which is not in the bootstrap sample, denoted as Out-Of-Bag (OOB) data, using the tree grown with the bootstrap sample.
- 2) Aggregate the OOB predictions. Measure the error rate, and define it as the OOB estimate of error rate.

For classification problems, given a set of simple trees and a set of random predictor variables, the RF method defines a margin function that measures the extent to which the average number of votes for the correct class exceeds the average vote for any other class presents in the dependent variable. This measure provides a convenient way of making predictions and associating a confidence measure with those predictions. For regression problems, RFs are formed by growing simple trees and each capable of producing a numerical response value. Here, the predictor set is randomly selected from the same distribution and for all trees. Given the above, the mean-square error for a Random Forest is given by [12]:

$$\text{Mean Error} = (\text{observed} - \text{tree response})^2$$

The predictions of the RF are taken to be the average of the predictions of the trees:

$$\text{Random Forest Predictions} = \frac{1}{K} \sum_{k=1}^K K^{th} \text{ tree response}$$

where the index  $k$  runs over the individual trees in the forest.

Random forests are a truly 'random' statistical method in which the model results can vary from one run to another run. Furthermore, as random forest builds independently, this makes their construction inherently parallel, allowing us the flexibility to exploit parallel computer architectures.

#### V. EXPERIMENTS AND RESULTS

In this research, we conduct our experiments on two fly wing image datasets and these datasets are consisted of equal number of the male and female fly wing images. The first

dataset, Dataset-1, contains 300 images and this dataset has 150 male and 150 female fly wing images. The second dataset, Dataset-2, contains 200 images and this dataset also has equal number of male and female fly wing images. These image datasets, supplied by Ian Dworkin (Dworkin Lab) at Michigan State University, are collected during a large scale experiment to examine how mutations of two biological signaling pathways influenced the wing shape [15], [16]. For each dataset, the wing images of the male and female flies are equally distributed into gallery and probe set. The gallery set is used to train SVM and RM classifiers and probe set is used for validation purpose. Each Image in the dataset is normalized to grey scale with a resolution of  $614 \times 266$  pixels and a horizontal orientation.

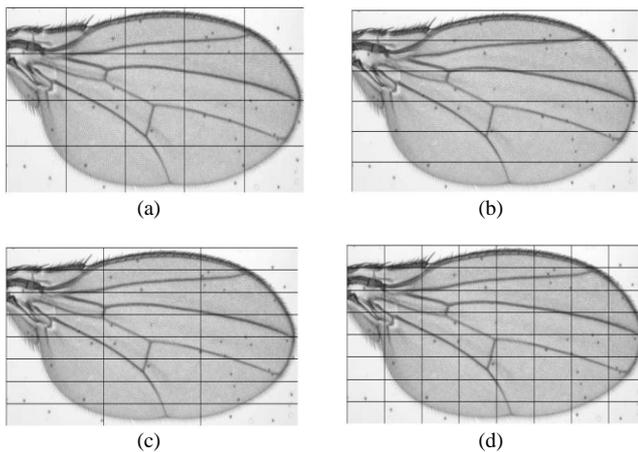


Fig. 3. Wings of male flies with different patches (a) 20 patches ( $4 \times 5$ ), (b) 6 patches ( $6 \times 1$ ), (c) 24 patches ( $8 \times 3$ ), (d) 64 patches ( $8 \times 8$ ).

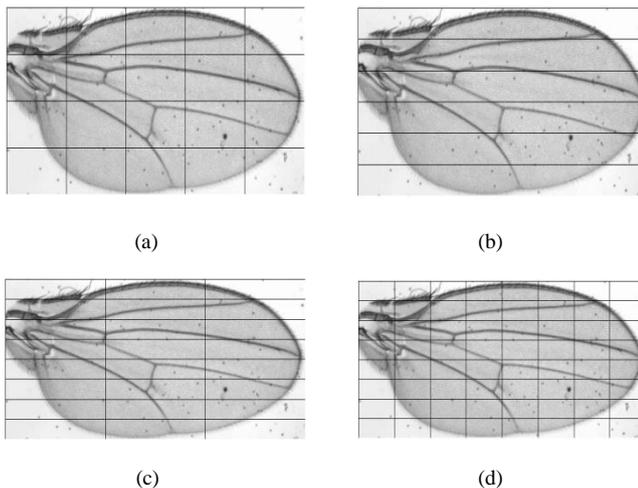


Fig. 4. Wings of female flies with different patches (a) 20 patches ( $4 \times 5$ ), (b) 6 patches ( $6 \times 1$ ), (c) 24 patches ( $8 \times 3$ ), (d) 64 patches ( $8 \times 8$ ).

In our research effort, we extract the fly wing texture features by using both the MLBP and LBP techniques. To extract more subtle and discriminating features, we have divided each image into different combinations of patches. Fig. 3 and Fig. 4 show some male and female sample images containing the different combination of patches. In feature extraction process, we employ only the uniform pattern as these uniform patterns correspond to important micro features such as edges, corners, spots as mentioned previously and these patterns cover the major proportion of the total number of patterns [4]. Each fly wing image patch is

represented by 59 sign and 59 magnitude components since eight ( $P = 8$ ) circularly surrounded neighbors are used. We then concatenate the sign and magnitude components and present a single patch by  $59 \times 2 = 118$  components. For example, with an image divided into  $4 \times 5$  patches (see Fig. 3(a)), we have  $4 \times 5 \times 118 = 2360$  features with MLBP approach which divides the images evenly in sign and magnitude vectors, while  $20 \times 59 = 1180$  sign features are obtained with the traditional LBP approach.

TABLE I: BEST MLBP PATCHES ON DATASET-1 USING SVM.

Patches (row $\times$ column)	Number of features	Accuracy %
$6 \times 1$	708	94
$8 \times 3$	2832	92.67
$2 \times 4$	944	91.33
$3 \times 4$	1416	91.33
$6 \times 4$	2832	91.33
$7 \times 3$	2478	91.33
$9 \times 7$	7434	91.33
$10 \times 1$	1180	91.33
$4 \times 7$	3304	90.67
$6 \times 3$	2124	90.67
$7 \times 1$	826	90.67

TABLE II: BEST LBP PATCHES ON DATASET-1 USING SVM.

Patches (row $\times$ column)	Number of features	Accuracy %
$4 \times 5$	1180	92.67
$8 \times 8$	3776	92.67
$9 \times 10$	5310	92.67
$2 \times 1$	118	92
$4 \times 7$	1652	92
$6 \times 4$	1416	92
$7 \times 8$	3304	92
$4 \times 4$	944	91.33
$5 \times 6$	1770	91.33
$6 \times 7$	2478	91.33
$4 \times 3$	708	90.67

Table I and II report the top eleven recognition accuracies for the MLBP and LBP methods, respectively on the Dataset-1 using the SVM. Similarly, Table III and Table IV include the results of top eleven best patch combinations for Dataset-2. We can see from Table I that MLBP achieves the best accuracy of 94%, with the  $6 \times 1$  patch combination, while the traditional LBP obtains the highest accuracy of 92.67% with the  $4 \times 5$ ,  $8 \times 8$  and  $9 \times 10$  patch combinations as reported in Table II on the Dataset-1 with SVM. Similarly, from Table III, we can see that for the  $4 \times 4$  patch arrangement, MLBP achieves the highest accuracy of 88% using the SVM on the Dataset-2. Table IV shows that  $4 \times 3$ ,  $4 \times 4$ ,  $7 \times 3$  and  $8 \times 3$  patch

combinations reveal the best accuracy of 86% with the traditional LBP on the Dataset-2 using the SVM. Therefore, it is found that MLBP outperforms the traditional LBP on both of the datasets. Table V shows the SVM classification accuracies obtained by using the four kernels and it can be seen clearly that the linear kernel archives the best accuracy of 94%. The reason seems to be that the size of the feature vector, used in this research, is huge as compared to number of samples per class.

For this experimental work, we consider different patches of every combination of 10 rows and 10 columns and calculate the average classification accuracy over all the observation outputs (see Table VI and Table VII). Table VI shows that average classification rates of different combination of patches for Dataset-1 are 89% and 89.50% for MLBP and LBP, respectively using SVM. However, for Dataset-1 with different patches, the best accuracy, we obtain is 94% using the MLBP and SVM. With the traditional LBP and SVM, the highest achieved accuracy is 92.67% on Dataset-1 as reported in Table VI. When the entire fly wing image is used for experiments, with SVM, the obtained recognition rates are 88% and 86.67% for MLBP and LBP, respectively. For the different combinations of patches on Dataset-2, the average SVM classification accuracies obtained using MLBP and LBP are 80.23% and 78%, respectively (see Table VII). On the Dataset-2, the best classification accuracies obtained with SVM are 88% and 86% for MLBP and LBP, respectively. We gain 76% and 74% accuracies for MLBP and LBP, respectively on the Dataset-2 when we consider the entire image for feature extraction using SVM. Best accuracy of 88% and average is 80.23% for MLBP feature extractor with SVM and with the LBP extractor, the highest recognition rate is 86% and average precision is 78%. From Table VI and Table VII, we obtain that MLBP with SVM outperforms the LBP in most of the cases when the patches are considered

To compare the classification results obtained using SVM with the RF, we use the different combination of patches and provide the wing features to RF in an effort to get the classification accuracies as shown in Table VI and Table VII. As the process is purely random, the accuracy rate varies in every run. We calculate the average rate over different patches and also on the entire image for different number of trees. For Dataset-1 with MLBP and RF scheme, we obtain 90% mean accuracy while the highest accuracy is 92% using different arrangement of patches and the average accuracy is 87% when the entire image is considered. For the traditional LBP with RF, the average and highest recognition rates are 88% and 91%, respectively on the Dataset-1 for different patch combinations. Without any patch combination, considering the complete image for feature extraction, we obtain a RF accuracy of 87% with both LBP and MLBP on the Dataset-1. On the Dataset-2, the highest accuracy with various patches is 87% and average accuracy is 84% for MLBP with RF algorithm. For original LBP and RF scheme, the average classification rate is 83% and highest rate is 87% for different patches. For Dataset-2, the RF accuracy with LBP is 83% when no patch is used. From Table VI and Table VII, we find that MLBP with RF outperforms the LBP in most of the cases when the patches are considered as similar

with MLBP and SVM scheme. It should be noted that in the first fly wing biometrics experiments, Shelton *et al.* [8] achieved an average recognition accuracy of 73.16% using GEFEML feature extractors on a different dataset of *Drosophila*.

TABLE III: BEST MLBP PATCHES ON DATASET-2 USING SVM

Patches (row × column)	Number of features	Accuracy %
4 × 4	1888	88
2 × 9	2124	86
5 × 10	5900	86
8 × 3	2832	86
4 × 5	2360	85
7 × 4	3304	85
10 × 3	3540	85
3 × 8	2832	84
7 × 1	826	84
4 × 1	472	84
6 × 7	4956	83

TABLE IV: BEST MLBP PATCHES ON DATASET-2 USING SVM

Patches (row × column)	Number of features	Accuracy %
4 × 3	708	86
4 × 4	944	86
7 × 3	1239	86
8 × 3	1416	86
2 × 3	354	84
2 × 8	944	84
7 × 4	1652	84
8 × 4	1888	84
5 × 3	885	83
1 × 3	177	82
10 × 1	590	82

TABLE V: CLASSIFICATION ACCURACY (%) OF DIFFERENT SVM KERNELS

SVM Kernel	Patch Combination	Accuracy (%)
Linear	6 × 1	94
RBF	3 × 3	64
Polynomial	2 × 1	86
Quadratic	2 × 4	59

TABLE VI: CLASSIFICATION ACCURACY (%) ON DATASET-1 USING DIFFERENT SCHEMES

		Dataset-1		
		With patches		Without patches (Entire image) Accuracy
		Avg. Accuracy	Highest Accuracy	
SVM	MLBP	89%	94%	88%
	LBP	89.50%	92.67%	86.67%
Random Forest	MLBP	90%	92%	87%
	LBP	88%	91%	87%

TABLE VII: CLASSIFICATION ACCURACY (%) ON DATASET-2 USING DIFFERENT SCHEMES

		Dataset-2		
		With patches		Without patches (Entire Image)
		Avg. Accuracy	Highest Accuracy	Accuracy
SVM	MLBP	80.23%	88%	76%
	LBP	78%	86%	74%
Random Forest	MLBP	84%	87%	84%
	LBP	83%	87%	83%

## VI. CONCLUSION

In this research approach, to classify the *Drosophila melanogaster* as either female or male using their wing's texture analysis, we explore the application of MLBP as a fly wing feature extractor. The MLBP extractor includes both the sign and magnitude components for robust feature elicitation. We also examine the performance of LBP and MLBP for multiple patch variations. Although the approach using LBP can provide accurate results, the proposed scheme with MLBP has proven to be consistently more precise than the traditional LBP. The SVM and RF algorithms are utilized to classify the gender of the fly wing. The performance of the proposed approach is validated on two datasets with an encouraging performance.

## ACKNOWLEDGMENT

The authors would like to thank the NSF for their support of this research.

## REFERENCES

- [1] D. J. Bornemann, J. E. Duncan, W. Staatz, S. Selleck, and R. Warrior, "Abrogation of heparan sulfate synthesis in *Drosophila* disrupts the Wingless, Hedgehog and Decapentaplegic signaling pathways," *Development*, vol. 131, no. 9, pp. 1927-1938, 2004.
- [2] C. Han, T. Y. Belenkaya, M. Khodoun, M. Tauchi, and X. Lin, "Distinct and collaborative roles of *Drosophila* EXT family proteins in morphogen signalling and gradient formation," *Development*, vol. 131, no. 7, pp. 1563-1575, April 2004.
- [3] Y. Takei, Y. Ozawa, M. Sato, A. Watanabe, and T. Tabata, "Three *Drosophila* EXT genes shape morphogen gradients through synthesis of heparan sulfate proteoglycans," *Development*, vol. 131, no. 1, pp. 73-82, January 2004.
- [4] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray scale and rotation invariant texture analysis with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, 2002.
- [5] Z. Guo, L. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Trans. on Image Processing*, vol. 19, no. 6, pp. 1657-1663, June 2010
- [6] Y. Guo, G. Zhao, and M. Pietikäinen "Discriminative features for texture description," *Pattern Recognition*, vol. 45, no. 10, pp. 3834-3843, 2012.
- [7] Y. Guo, G. Zhao, and M. Pietikäinen "Texture classification using a linear configuration model based descriptor," in *Proc. the British Machine Vision Conference (BMVC 2011)*, Dundee, UK, 119.1-119.10.
- [8] M. Payne, G. Dozier, J. Shelton, J. Turner, J. Adams, J. Carter, H. Williams, C. Hansen, and I. Dworkin, "Fly wing biometrics using genetic and evolutionary feature extraction," *IEEE Symposium Series on Computational Intelligence*, 2003.
- [9] J. Shelton, K. Bryant, S. Abrams, L. Small, J. Adams, D. Leflore, and

- G. Dozier, "Genetic and evolutionary biometric security: disposable feature extractors for mitigating biometric replay attacks," *Procedia Computer Science*, vol. 8, pp. 351-360.
- [10] C. Cortes and V. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, pp. 273-297, 1995.
- [11] B. Schölkopf, K. Tsuda, and J.P. Vert, "Kernel Methods in Computational Biology," MIT Press series on Computational Molecular Biology. MIT Press, 2004.
- [12] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.
- [13] V. Ghosal, P. Tikmani, and P. Gupta, "Face Classification Using Gabor Wavelets and Random Forest," in *Proc. 2009 Canadian Conference on Computer and Robot Vision*, pp. 68-73, 2009.
- [14] T. Ho, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 832-844, 1998.
- [15] I. Dworkin and G. Gibson, "Epidermal growth factor receptor and transforming growth factor-beta signaling contributes to variation for wing shape in *Drosophila melanogaster*," *Genetics*, vol. 173, no. 3, pp. 1417-1431.
- [16] V. Debat, A. Debelle, and I. Dworkin, "Plasticity, canalization, and developmental stability of the *Drosophila* wing: joint effects of mutations and developmental temperature," *Evolution*, vol. 63, no. 11, pp. 2864-2876, Nov. 2009, doi: 10.1111/j.1558-5646.2009.00774.x.



Foysal Ahmad is currently a MS student at the Department of Computer Science, North Carolina A&T State University, North Carolina, U.S.A. He received his B.Sc. degree from Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh. He also worked at Samsung R&D Institute, Dhaka, Bangladesh as software engineer. His research interest includes biometrics, computer vision, image processing, pattern recognition, and machine learning.



Kaushik Roy received his PhD from Concordia University, Montreal, QC, Canada in 2011 in Computer Science. He also completed his MS degree in Computer Science from the Concordia University in 2006 and B.Sc. degree in Computer Science from University of Rajshahi, Bangladesh in 2001. Kaushik Roy is currently an assistant professor at the Department of Computer Science, and assistant director of the Center for Advanced Studies in Identity Sciences (CASIS), North Carolina A&T State University, USA. Previously, he worked as a postdoctoral fellow in the Department of Electrical and Computer Engineering, University of Waterloo, ON, Canada during 2011-2012. He also taught at Rajshahi University of Engineering and Technology (RUET) as a lecturer of the Department of Computer Science and Engineering during 2001-2004. He is also the recipient of several fellowships and awards including the prestigious NSERC Visiting Fellowship, FQRNT B3 (Postdoctoral), NSERC (Doctoral) and FQRNT B2 (Doctoral). His research interests include Biometrics, cyber identity, game theory, information fusion, computer vision, machine learning, and pattern recognition. He has published 1 book, 2 book chapters, 11 journal articles and 38 conference articles.



Brian O'Connor is an undergraduate student at North Carolina Agricultural and Technical State University, Greensboro, NC, USA, in the Computer Science Department. Brian is scheduled to graduate with his bachelor's degree in computer science in spring 2014. He is currently working as a research assistant at North Carolina A&T and has been in his position for a year. His research interest includes Cyber Identity, Biometrics and Machine Learning.



Joseph Shelton is a doctoral student at North Carolina Agricultural and Technical State University, Greensboro, NC, USA, in the Computer Science Department. Joseph obtained both his bachelor's degree and master's degree in computer science at North Carolina A&T State University in 2010 and 2012.

He is currently working as a research assistant at North Carolina A&T and has done so for the last three

years. He has helped publish a book chapter in 'New Trends and Developments in Biometrics' titled 'Genetic and Evolutionary Biometrics' and has published over 20 articles in the field of biometrics and genetic & evolutionary computations. Joseph's research interests include biometrics, cyber security and evolutionary computation.

Mr. Joseph Shelton has received an award for 1<sup>st</sup> runner up for best student paper at the Conference on Systems Engineering Research (CSER 2012).



**Gerry Vernon Dozier** is a professor and the chair of the Computer Science Department at North Carolina A&T State University. He is the director of the Center for Advanced Studies in Identity Sciences (CASIS), as well as the PI for the Center for Cyber Defense (recognized by the National Security Agency and the Department of Homeland Security as a Center for Academic Excellence in Information Assurance Education). During Gerry's tenure as chair, the

department has seen an increase in extramural funding and research publications as well as the establishment of a Ph.D. program. He has also lead in the development of an undergraduate research program where approximately 20% of the undergraduate students are active participants in funded research projects. Under Gerry's leadership, the NSF Alliance for the Advancement of African American Researchers in Computing (A4RC, [www.a4rc.org](http://www.a4rc.org)) experienced a threefold increase (from 6 to 20) in the number

of participating universities. A4RC was effective in increasing the number of African-American recipients of advanced degrees in Computer Science.

Gerry has published over 130 conference and journal publications. He has served as an Associate Editor of the IEEE Transactions on Evolutionary Computation and the International Journal of Automation & Soft Computing. Gerry is also a member of the Editorial Board for the International Journal of Intelligent Computing & Cybernetics. His research interests include: Artificial & Computational Intelligence, Genetic, Evolutionary, and Neural Computing, Biometrics, Identity Sciences, Cyber Identity, Distributed Constraint Reasoning, Artificial Immune Systems, Machine Learning and Network Intrusion Detection. Gerry earned his Ph.D. from North Carolina State University.



**Ian Dworkin** was born in Montreal, Canada. He studied biochemistry as an undergraduate at Trent University, followed by a PhD studying genetics at the University of Toronto. After completing work as an NSERC post-doctoral researcher at North Carolina State University and the University of North Carolina at Chapel Hill he joined the faculty of Michigan State University in the Program of Ecology, Evolutionary Biology and Behavior where he is currently an associate professor. His interests

include evolutionary genetics of shape (geometric morphometrics), behaviour and genomics.