CLASSIFICATION OF DROSOPHILA EMBRYONIC DEVELOPMENTAL STAGE RANGE BASED ON GENE EXPRESSION PATTERN IMAGES

Jieping Ye [a,b*] Jianhui Chen [a,b] Qi Li [c] Sudhir Kumar [a,d]

[a] *Center of Evolutionary Functional Genomics, Biodesign Institute, Arizona State University, Tempe, AZ 85287-5301. {jieping.ye,jianhui.chen,s.kumar}@asu.edu.*

[b] *Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287.*

[c] *Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716. qili@cis.udel.edu.*

[d] *School of Life Sciences, Arizona State University, Tempe, AZ 85287.*

The genetic analysis of spatial patterns of gene expression relies on the direct visualization of the presence or absence of gene products (mRNA or protein) at a given developmental stage (time) of a developing animal. The raw data produced by these experiments include images of the *Drosophila* embryos showing a particular gene expression pattern revealed by a gene-specific probe. The identification of genes showing spatial and temporal overlaps in their expression patterns is fundamentally important to formulating and testing gene interaction hypotheses. Comparison of expression patterns is most biologically meaningful when images from a similar time point (developmental stage range) are compared. In this paper, we propose a computational system for automatic developmental stage classification by image analysis. This classification system uses image textural properties at a sub-block level across developmental stages as distinguishing features. Gabor filters are applied to extract features of image sub-blocks. Robust implementations of Linear Discriminant Analysis (LDA) are employed to extract the most discriminant features for the classification. Experiments on a collection of 2705 expression pattern images from early stages show that the proposed system significantly outperforms previously reported results in terms of classification accuracy, which shows high promise of the proposed system in reducing the time taken by biologists to assign the embryo stage range.

1. INTRODUCTION

Gene expression in a developing embryo is modulated in particular cells in a time-specific manner, which leads to the differentiation of cell fates. Research efforts into the spatial and temporal characteristics of gene expression patterns of the model organism *Drosophila melanogaster* (the fruit fly) have been at the leading-edge of scientific investigations into the fundamental principles of animal development.[5, 16] These studies have now established that the same gene (or its product) may be utilized in different ways at different times during development, and that multiple genes show similar expression patterns in one or more developmental stages. The genetic analysis of spatial patterns of gene expression relies on the direct visualization of the presence or absence of gene products (mRNA or protein) at a given developmental stage (time) of a developing animal. The raw data produced from these experiments includes images of the *Drosophila* embryo showing a particular gene expression pattern revealed by a gene-specific probe. The knowledge of the spatial overlap of patterns of gene expression is important to understanding the interplay of genes in different stages of development.[5, 16]

Estimation of the pattern overlap is most biologically meaningful when images from a similar time point (developmental stage range) are compared. Stages in *Drosophila melanogaster* development denote the time after fertilization at which certain, specific events occur in the developmental cycle. Embryogenesis is traditionally divided into a series of consecutive stages distinguished by morphological markers.[1] The duration of developmental stages varies from 15 minutes to more than 2 hours; therefore, the stages of development are differentially represented in the embryo collections. Some consecutive stages, although morphologically distinguishable, differ very little in terms of changes in gene expression, whereas other stage transitions, such as the onset of zygotic transcription or organogenesis, are accompanied by massive changes in gene expression.[1] The first 16 stages of embryogenesis are divided into six convenient stage ranges (stages 1-3, 4-6, 7-8, 9-10, 11-12 and 13-16). In recent high throughput experiments,[18] each image is assigned to one of the stage ranges manually.

*Corresponding author.

In this paper, we examine how image analysis can be used for automatic stage range determination (classification). In order to distinguish between different stage ranges of development, we need to use embryo morphology to extract features. Across the various developmental stages, a distinguishing feature is image textural properties at a sub-block level, because image texture at the sub-block level changes as embryonic development progresses (Fig. 1). The staining procedure helps illuminate the morphological features of the transparent embryos as well. We thus apply Gabor filters[7] to extract the textural features of image sub-blocks. Since not all features are useful for stage range discrimination, we apply robust implementations of Linear Discriminant Analysis (LDA)[8, 10, 14] for the extraction of the most discriminant features, which are linear combinations of the textural features derived from the Gabor filters. Finally, the Nearest-Neighbor (NN) algorithm and Support Vector Machines (SVM)[4, 6, 19] are employed for classification (stage range determination). Our experiments on a collection of 2705 expression pattern images from early stages show that the proposed system achieves about 86% accuracy, when less than 10% of the data is used for the training, which is significantly higher than previously reported result (about 73%).[11]

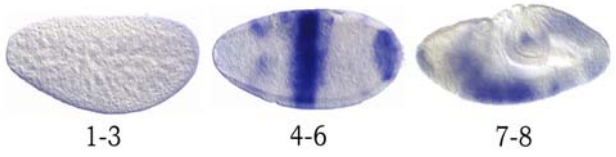

Fig. 1. Spatial and temporal view of *Drosophila* images across different stages (1–8) of development (of the same gene Kr). The figure shows the morphological changes at the anterior and posterior end of the embryo during stages 4-6 and the morphological changes in the middle regions of the embryo during stages 7–8. The textural features (based on the morphology of the embryo) are different from the gene expression, which is indicted by the blue staining.

1.1. Raw image pre-processing

We used a collection of 2705 embryo images from three different developmental stage ranges (1–3, 4–6, and 7–8) in our study. The raw images of *Drosophila Embryo* were collected from the Berkeley Drosophila Genome Project (BDGP).[18] Gene expression pattern images were in different sizes and orientations. The image standardization procedure from Ref. 16 was applied and all images were standardized to the size of 128×320.

Next, we applied *Histogram Equalization*[13] to improve the contrast and obtain approximately an uniform histogram distribution, while still keeping the detailed information for the processed images.

Finally, we applied *Gabor Filters*[7] to extract the textural features of image sub-blocks. Gabor Filters are well-known for texture analysis,[2] as they are effective in extracting information in different spatial frequency ranges and orientations. We found the textural features obtained via *Gabor Filters* very effective in stage range classification (see Section 4). The number of textural features extracted via Gabor Filters is 384. Since not all features are useful for stage discrimination, we applied the Linear Discriminant Analysis (LDA) for extracting the most discriminant features before the classification.

2. LINEAR DISCRIMINANT ANALYSIS

Linear Discriminant Analysis (LDA)[8, 10, 14] is a well-known method for feature extraction that projects high-dimensional data onto a low-dimensional space to maximize class separability. The optimal projection or transformation in classical LDA is obtained by minimizing the within-class distance and maximizing the between-class distance simultaneously, thus achieving maximum class discrimination.

Given a training dataset consisting of n data points (images), $\{a_i\}_{i=1}^{n} \in \mathbb{R}^m$, from k different classes, classical LDA aims to compute the transformation $G \in \mathbb{R}^{m \times \ell}$ $(\ell < m)$ that maps a_i to a vector y_i in the ℓ-dimensional space as follows:

$$G : a_i \in \mathbb{R}^m \rightarrow y_i = G^T a_i \in \mathbb{R}^\ell.$$

In classical LDA, the transformation matrix G is computed so that the class structure is preserved. The class structure is quantified by three scatter matrices, called the *within-class scatter* S_w, the *between-class scatter* S_b, and the *total scatter* S_t, defined below.

Assume that there are k classes in the dataset. Suppose c_i, S_i, n_i are the centroid, covariance matrix, and sample size of the i-th class, respectively,

and c is the global centroid. Define the matrices

$$H_w = \frac{1}{\sqrt{n}}[(A_1 - c_1 e^T), \cdots, (A_k - c_k e^T)], \quad (1)$$

$$H_b = \frac{1}{\sqrt{n}}[\sqrt{n_1}(c_1 - c), \cdots, \sqrt{n_k}(c_k - c)], \quad (2)$$

$$H_t = \frac{1}{\sqrt{n}}(A - ce^T), \quad (3)$$

where $A = [x_1, \cdots, x_n]$ is the data matrix, A_i is the data matrix of the i-th class, n_i is the size of the i-th class, and e is the vector of all ones. Then the three scatter matrices are defined as follows:[10]

$$S_w = H_w H_w^T,\ S_b = H_b H_b^T,\ \text{and}\ S_t = H_t H_t^T.$$

It follows from the definition that trace(S_w) measures the within-class cohesion, trace(S_b) measures the between-class separation, and trace(S_t) measures the variance of the dataset, where the trace[12] of a square matrix is the summation of all its diagonal entries . It is easy to verify that $S_t = S_b + S_w$.

The scatter matrices in the reduced space (projected by G) are $G^T S_w G$, $G^T S_b G$, and $G^T S_t G$, respectively. The optimal transformation G in classical LDA is computed by maximizing the following objective function:[8, 10, 14]

$$f_1(G) = \text{trace}\left(\left(G^T S_w G\right)^{-1} G^T S_b G\right), \quad (4)$$

subject to the constraint that $G^T S_w G = I_\ell$, where I_ℓ is the identity matrix of size ℓ. The optimal solution is given by the eigenvectors of $S_w^{-1} S_b$ corresponding to the nonzero eigenvalues, provided that S_w is nonsingular. Since $S_t = S_b + S_w$, the solution can also be obtained by computing the eigenvectors of $S_t^{-1} S_b$, assuming S_t is nonsingular. The reduced dimension, ℓ, is no larger than $k-1$, where k is the number of classes, as the rank of S_b is bounded from above by $k-1$. In practice, ℓ often equals $k-1$. Note that the total scatter matrix is a multiple of the sample covariance matrix and is required to be nonsingular. If a small number of expression pattern images is used in the training set, all scatter matrices in question can be singular. This is known as the *singularity* or *undersampled* problem.[15]

We have recently developed Uncorrelated LDA (ULDA)[21] as an extension of classical LDA. A key property of ULDA is that the features in the transformed space of ULDA are uncorrelated to each other, thus reducing the redundancy in the transformed (dimension reduced) space. Furthermore, ULDA is applicable, even when all scatter matrices are singular, thus overcoming the singularity problem. The optimal transformation G of ULDA can be computed by maximizing the following objective function:

$$f_2(G) = \text{trace}\left((G^T S_t G)^+ G^T S_b G\right), \quad (5)$$

subject to the constraint that $G^T S_t G = I_\ell$, where M^+ denotes the pseudo-inverse[12] of a matrix M. The computation of the optimal transformation of ULDA is based on the simultaneous diagonalization of the three scatter matrices.[21] Let X be the matrix that simultaneously diagonalizes S_b, S_w, and S_t. That is,

$$X^T S_b X = D_b, X^T S_w X = D_w,\ \text{and}\ X^T S_t X = D_t, \quad (6)$$

where D_b, D_w, and D_t are diagonal, and the diagonal entries of S_b are sorted in the non-increasing order. Then $G = X_q$ solves the optimization problem in Eq. (5), where X_q consists of the first q columns of X with $q = \text{rank}(S_b)$.

ULDA has been applied successfully in several applications, including microarray gene expression data analysis.[22] However, we have observed that for data containing large amount of noises, ULDA has been shown to be less effective.[21] We employ the regularization technique to improve the robustness of ULDA. The algorithm is called Regularized ULDA (RULDA). Regularization is commonly used to stabilize the sample covariance matrix estimation and improve the classification performance.[9] Regularization is also the key to many other machine learning methods such as Support Vector Machines (SVM),[19] spline fitting,[20] etc. In RULDA, a regularization parameter λ is added to the diagonal elements of the total scatter matrix S_t as $S_t + \lambda I_m$, where I_m is the identity matrix of size m. The optimal transformation G of RULDA is given by computing the eigenvectors of

$$(S_t + \lambda I_m)^{-1} S_b. \quad (7)$$

The performance of RULDA is critically dependent on the estimation of an appropriate regularization value λ, because a large λ may significantly disturb the information on S_t, while a small λ may not be

effective enough to solve the singularity problem. Cross-validation is commonly used to estimate the optimal λ from a finite set,

$$\Lambda = \{\lambda_1, \cdots, \lambda_N\},$$

of N candidates. We used $N = 100$ in our experiments.

With the discriminant features extracted via LDA, the Nearest-Neighbor (NN) algorithm and Support Vector Machines (SVM) are applied for classification.

3. K-NEAREST NEIGHBOR AND SUPPORT VECTOR MACHINES FOR CLASSIFICATION

K-Nearest Neighbor (KNN)[8, 14] is a non-parametric classifier and theoretical proofs have shown that its error is asymptotically at most 2 times of the Bayesian error rate. KNN finds the K nearest neighbors among training samples based on a certain distance measure, and uses the categories of the K neighbors to determine the category of the test sample. The parameter K for the number of neighbors can be selected by cross-validation. In our experiments, K is set to be 1 and the algorithm is called Neighbor-Neighbor (NN).

Support Vector Machines (SVM)[3, 6, 19] are the state-of-the-art classifiers for many classification problems.[3] SVM finds a maximum margin separating hyperplane between two classes. It leads to a straightforward learning algorithm that can be reduced to a convex optimization problem. The formulation can be extended to multi-class classifications.[6, 17] SVM is attractive due to its well developed theory.[19] Another appealing feature of SVM classification is the sparseness of its representation of the decision boundary. The maximum margin hyperplane can be represented as a linear combination of data points. Those training examples that receive nonzero weights, are called the *support vectors*, since removing them would change the location of the separating hyperplane. Kernels[6, 17] can be used to extend SVM to classify nonlinearly separable data. We apply linear SVM in our experiments.

4. RESULTS AND DISCUSSIONS

In this section, we experimentally evaluate the proposed system on embryonic developmental stage range classification. A collection of 2705 embryo images from three different developmental stage ranges (1–3, 4–6, 7–8) was used in our study.

We performed our study by a random splitting of the whole dataset into training and test sets. The dataset was partitioned randomly into a training set consisting of n images (n denotes the training sample size) and a test set consisting of the remaining $2705 - n$ images. We varied the training sample size n from 30 to 540. To reduce the variability, the splitting was repeated 50 times and the resulting accuracies were averaged.

We first examined the effect of Histogram Equalization (HE) and Gabor Filters (GF) on stage range classification. To this end, we ran the experiments under four different conditions: "NO" without any pre-processing, "HE" with Histogram Equalization, "GF" with Gabor Filters, and "HE+GF" with both Histogram Equalization and Gabor Filters. The classification result (accuracy in percentage) using SVM as the classifier is shown in Table 1. We can observe that both the HE and GF operations are effective in classification, while GF is more effective than HE. In the following experiments, all images were preprocessed via both operations.

Table 1. Effect of image pre-processing operations on stage range classification (accuracies shown in percentage). NO: No pre-processing; HE: Histogram Equalization; and GF: Gabor Filters.

size	Image pre-processing operation			
n	NO	HE	GF	HE + GF
30	53.49	61.22	67.91	76.31
60	62.04	65.27	77.77	80.34
90	67.17	68.21	79.91	82.73

Next, we evaluated the proposed system on stage range classification. We employed both RULDA and ULDA to extract discriminant features before applying NN and SVM for classification. We can observe from Table 2 that RULDA plus NN and RULDA plus SVM achieve the best overall performance. When less than 10% of the images is used in the training set, they achieve about 86% accuracy, which is

significantly higher than previously reported result[11] (about 73%). The key feature of the proposed computational system in comparison with the previous work is the inclusion of the feature extraction step via Regularized ULDA (RULDA), as well as the use of SVM as the classifier. Experimental results in Table 2 show the effectiveness of both RULDA and SVM for stage range classification.

Table 2. Comparison of two feature extraction algorithms (ULDA and RULDA) and two classifiers (NN and SVM) on classification accuracy and standard deviation (in parenthesis) in percentage.

training sample size n	ULDA		RULDA	
	NN	SVM	NN	SVM
30	76.94	76.94	76.55	76.31
	(3.48)	(3.48)	(4.33)	(4.18)
60	79.89	79.89	80.91	80.34
	(2.33)	(2.33)	(3.08)	(2.93)
90	80.68	80.68	82.71	82.73
	(2.11)	(2.11)	(3.09)	(2.29)
180	77.22	77.22	85.74	86.10
	(2.62)	(2.62)	(1.82)	(1.65)
300	66.30	66.30	86.60	87.37
	(2.67)	(2.67)	(1.18)	(1.58)
480	68.29	68.69	87.48	88.75
	(2.19)	(2.39)	(0.99)	(1.16)
540	73.80	73.90	87.24	88.91
	(2.01)	(2.20)	(1.26)	(1.16)

In general, as the training sample size, n, increases, the classification accuracy of both RULDA plus NN and RULDA plus SVM increases. We observe that ULDA does not perform well when the training sample size n is large. The rationale behind this may be that ULDA involves the minimum redundancy (uncorrelated features) in the transformed space and is susceptible of overfitting. The expression pattern images may contain a large amount of noises due to the errors encountered in high throughput experiment and in image pre-processing. RULDA significantly improves ULDA in these cases, which shows the effectiveness of the regularization applied in RULDA. The regularization parameter in RULDA is estimated via cross-validation using the training data. When the training set is large, the estimation of the regularization value is more reliable and more robust to the noise. This explains the relatively larger difference between RULDA and ULDA in classification, when the training sample size n is large. Overall, RULDA plus SVM performs slightly better than RULDA plus NN, especially when the training sample size n is large.

Recall that RULDA projects the data onto $\mathbb{R}^{k-1}$, where k is the number of classes in the dataset. There are $k = 3$ stage ranges (classes) in our experiments, and all images are projected onto a 2D plane. To examine the effectiveness of the projection, we ran RULDA on a training set of 180 images and applied the projection to a test set of 2525 images. In Fig. 2, we showed the projection of a subset of test images (for clarity of presentation). We depicted each test image by the corresponding stage range (1, 2, and 3). Overall, the three stage ranges were separated well, which shows that the discriminant features derived via RULDA is effective in stage range discrimination. We observe that stage ranges 1 and 2 are connected, as well as stage ranges 2 and 3, while stage ranges 1 and 3 are better separated. Note that the embryonic development is a continuous process, where the cutting points (boundaries) between different stages are assigned manually. These are consistent with the data distribution (after projection) as shown in Fig. 2.

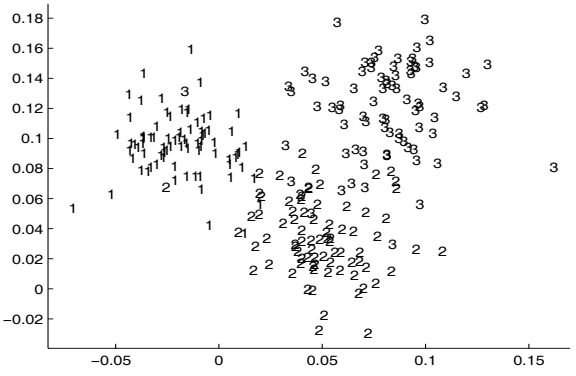

Fig. 2. Visualization of a subset of test images after the projection onto the 2D plane via RULDA. Images from the first range (1–3), the second range (4–6), and the third range (7–8) are depicted by "1", "2", and "3",respectively.

5. CONCLUSIONS

We present in this paper a computational system for automatic developmental stage classification by image analysis. This classification system applies Ga-

bor filters to extract textural features of image subblocks. Uncorrelated LDA (ULDA) and Regularized ULDA (RULDA) are employed to extract the most discriminant features for the classification. Experiments on a collection of 2705 expression pattern images from early stages show that the proposed system significantly outperforms previously reported results in terms of classification accuracy. The experimental results demonstrate the promise of the proposed computational system for embryonic developmental stage range classification. As a future work, we plan to test the proposed system using a much larger collection of expression pattern images including images from all stage ranges.

Acknowledgement

This research is sponsored by the Center of Evolutionary Functional Genomics of the Biodesign Institute at the Arizona State University and the National Institutes of Health (NIH).

References

1. M. Bownes. A photographic study of development in the living embryo of Drosophila melanogaster. *J Embryol Exp Morphol*, 33:789–801, 1975.
2. B.S.Manjunath and W.Y.Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.
3. C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
4. C. J. C. Burges. Geometric methods for feature extraction and dimensional reduction - a guided tour. *The Data Mining and Knowledge Discovery Handbook*, pages 59–92, 2005.
5. S.B. Carroll, J.K. Grenier, and S.D. Weatherbee. *From DNA to diversity : molecular genetics and the evolution of animal design. 2nd ed.* Malden, MA: Blackwell Pub, 2005.
6. N. Cristianini and J. Shawe-Taylor. *Support Vector Machines and other Kernel-based Learning Methods.* Cambridge University Press, 2000.
7. J.G. Daugman. Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression. *IEEE Transcations on Acoustics, Speech, and Signal Processing*, 36(7):1169–1179, 1988.
8. R.O. Duda, P.E. Hart, and D. Stork. *Pattern Classification.* Wiley, 2000.
9. J.H. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(405):165–175, 1989.
10. K. Fukunaga. *Introduction to Statistical Pattern Classification.* Academic Press, San Diego, California, USA, 1990.
11. M. Gargesha, J. Yang, B. Van Emden, S. Panchanathan, and S. Kumar. Automatic annotation techniques for gene expression images of the fruit fly embryo. In *Proceedings of SPIE (Visual Communications and Image Processing)*, pages 576–583, 2005.
12. G. H. Golub and C. F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, USA, third edition, 1996.
13. R.C. Gonzalez and R.E. Woods. *Digital Image Processing, Second Edition.* Addison-Wesley, 1993.
14. T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning : Data Mining, Inference, and Prediction.* Springer, 2001.
15. W.J. Krzanowski, P. Jonathan, W.V McCarthy, and M.R. Thomas. Discriminant analysis with singular covariance matrices: methods and applications to spectroscopic data. *Applied Statistics*, 44:101–115, 1995.
16. S. Kumar, K. Jayaraman, S. Panchanathan, R. Gurunathan, A. Marti-Subirana, and S.J. Newfeld. BEST: A novel computational approach for comparing gene expression patterns from early stages of Drosophila melanogaster development. *Genetics*, 162(4):2037–2047, 2002.
17. B. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond.* MIT Press, 2002.
18. P. Tomancak et al. Systematic determination of patterns of gene expression during Drosophila embryogenesis. *Genome Biol*, 3(12):research0088.1–14, 2002.
19. V.Vapnik. *The Nature of Statistical Learning Theory.* Springer, 1995.
20. G. Wahba. *Spline models for observational data.* Society for Industrial & Applied Mathematics, 1998.
21. J. Ye. Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems. *Journal of Machine Learning Research*, 6:483–502, 2005.
22. J. Ye, T. Li, T. Xiong, and R. Janardan. Using uncorrelated discriminant analysis for tissue classification with gene expression data. *IEEE/ACM Trans. Computational Biology and Bioinformatics*, 1(4):181–190, 2004.