TOWARDS AN IMAGE ANALYSIS TOOLBOX FOR HIGH-THROUGHPUT *DROSOPHILA* EMBRYO RNAI SCREENS

Ryan A. Kellogg[1,2]*, Amina Chebira*[1]*, Anupam Goyal*[3]*, Philip A. Cuadra*[2]*,*
Stefan F. Zappe[1]*, Jonathan S. Minden*[3] *and Jelena Kovačević*[1,2]

[1] Dept. of BME and Center for Bioimage Informatics, [2] Dept. of ECE, [3] Dept. of Biol. Sci.
Carnegie Mellon University, Pittsburgh, PA

ABSTRACT

We build an image analysis toolbox for high-throughput *Drosophila* embryo RNAi screens. The goal is to tag the embryo as normal, developmentally delayed or abnormal based on the ventral furrow formation. We break the problem into two parts: in the first, we detect the developmental stage based on the progress of the ventral furrow formation, and in the second, we tag the embryo as normal/developmentally delayed/abnormal based on the stage detected and the elapsed time. The crux of the algorithm is the multiresolution classifier, and we show that, by classifying in multiresolution spaces, we obtain better results than by classifying the embryo image alone. The final 2D accuracy obtained was 93.17%, while by using 3D information, it increased to 98.35%.

***Index Terms*—** High-throughput, screening, Drosophila, classification, multiresolution

1. STUDY OF *DROSOPHILA* EMBRYOS

The genome projects have brought unprecedented opportunities to understand molecular mechanisms of development and disease. The *Drosophila* sequences are of special interest because the fly serves as an important model organism for developmental and cellular processes common to higher eukaryotes, including humans. Comparative genomics studies have revealed that D. melanogaster, for example, has orthologs to 177 out of 289 examined human disease genes [1]. The genome sequence of D. melanogaster was published in 2000 [2], followed by the sequence of *Drosophila* pseudoobscura in 2005 [3].

While the *Drosophila* genome projects provide us with a wealth of data, the determination of the functions of the genes that are inferred from these sequences (approximately 13,600 genes for D. melanogaster) requires novel, highly efficient and high-throughput screening methods [4] and methods for automated phenotype analysis [5].

RNA interference (RNAi) is one such method that can be used to silence a specific gene in a cell or an organism [6]. Analysis of a change in phenotype due to gene silencing indicates the function of the silenced gene. Silencing a gene in an entire fly embryo through RNAi requires injection of embryos with designed, double-stranded RNA (dsRNA) early in embryonic development, prior to the formation of the syncytial blastoderm. A powerful MEMS-based system for automated, high-throughput injection of *Drosophila* embryos has been recently proposed [7]. Phenotype analysis after gene silencing is greatly facilitated through genetic engineering of *Drosophila* embryos that express, for example, green fluorescent protein (GFP) in a tissue of interest [8]. Modern confocal laser scanning fluorescence microscopes are capable of automatically acquiring image z-stacks of entire embryos with acceptable resolution within seconds, enabling time-lapse recording of fluorescently marked features of a large number of embryos. However, manual interpretation of the huge amount of generated 4D image data is impractical and error-prone. Fast, reliable, flexible, and efficient algorithms for automated 4D image analysis and phenotype detection are needed to enable high-throughput functional genomics screens.

To demonstrate the feasibility of automated image data analysis in *Drosophila* embryo RNAi screens, we have developed an algorithm for screening during early embryonic development based on ventral furrow formation. Ventral furrow formation is a key morphogenetic event during *Drosophila* gastrulation that leads to the internalization of mesodermal precursors [9]. We have trained image analysis algorithms to recognize the beginning, a middle stage and the end of ventral furrow formation of wildtype embryos. In subsequent experiments, we have then silenced specific genes, known to be implicated in ventral furrow formation, through the RNAi mechanisms. Our algorithms were able to detect phenotypes due to gene silencing based on deviations from the normal occurrence of ventral furrow formation stages over time.

2. AUTOMATED DETERMINATION OF DEVELOPMENTAL STAGE

2.1. Problem Statement

The problem we are addressing is that of labeling an image as normal/developmentally delayed/abnormal, given as input the image itself and the time stamp. We do this by first detecting the developmental stage based on the progress of the ventral furrow formation, and then, tagging the embryo as normal/developmentally delayed/abnormal based on the stage detected and the elapsed time.

2.2. Algorithm Details

We approach the first part of the problem, that of detecting the developmental stage as a classification problem. That is, we aim to design a map from the *signal space* of embryo images $\mathcal{X} \subset \mathcal{R}^{N \times N}$, ($N \times N$ is the image size) to a *response space* $\mathcal{Y} \subseteq \{$stage 1, stage 2, stage 3$\}$ of class labels. Thus, decision d is the map, $d\colon \mathcal{X} \mapsto \mathcal{Y}$ that associates an input image with a class label [10]. To reduce the dimensionality of the problem, one sets up a feature space $\mathcal{F} \subset \mathcal{R}^k, k \leq N^2$, between the input space and the response space. The feature extractor θ is the map $\theta\colon \mathcal{X} \mapsto \mathcal{F}$, and the classifier ψ is the map $\psi\colon \mathcal{F} \mapsto \mathcal{Y}$. The goal is to find a (θ, ψ) pair that maximizes the

This work was supported in part by NSF through awards 0515152 and ITR-EF-0331657, as well as the PA State Tobacco Settlement, Kamlet-Smith Bioinformatics Grant.

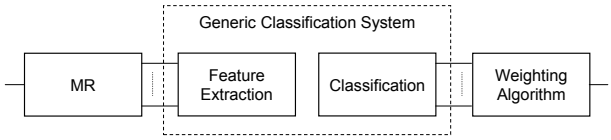

Fig. 1. The generic classification system (GCS) consists of feature extraction followed by classification (inside the dashed box). We add an MR block in front of GCS and compute features in MR subspaces (subbands). Classification is then performed on each of the subbands yielding local decisions which are then weighed and combined to give a final decision.

classification accuracy. A generic classification system (GCS) thus first computes the features and then classifies, as in the dashed box in Figure 1.

In [11], we introduced a concept of multiresolution (MR) classification for classification of protein subcellular location images, arguing that the nature of such images requires tools which offer localization in space and frequency as well as adaptivity. Thus, we classify in MR subspaces as opposed on the original image itself with the idea is that certain features will react well at a certain scale but not at another. We add an *MR Block* in front of the GCS, as in Figure 1. Given that now each subspace voices an opinion, these opinions are combined via a weighting algorithm.

MR Block. The basic MR block is the so-called *two-channel filter bank*. It, and its extensions, can be used to build decompositions, *wavelet packets* [12], custom-tailored to the image at hand. This is done by using this filter bank in a tree, iterating on any of the two-channels and its children. Moreover, the filter bank can have more then two channels, and can have more channels than the sampling factor (leading to redundant representations), etc.

Amongst the possible trees that one can use to analyze an image, the wavelet packets mentioned previously adapt themselves to the image at hand. However, this is possible only if a suitable *cost function* is available. That is, in order to adaptively build the tree, we need to find a suitable measure that will indicate whether a subband (a node in the tree) contains useful information or not. If it does, then we keep the node, otherwise, we prune it. Adaptive flavors of MR have been explored for their utility in classification in various domains [13]. These studies have used the transform domain coefficients themselves as features and so had a natural cost function in selecting the tree most adapted to the signal. In [14], we used wavelet packets for fingerprint identification and verification with remarkable results. As we do not have a natural cost measure, we simulate wavelet packets by decomposing fully and using all the nodes in the classification by attaching weights to each node, as described shortly. This process builds a decomposition adapted to the data set as it weighs higher the subbands with the high discriminative power, and lower those with the low discriminative power.

Feature Extraction Block. If using more than one feature set, we allow each one its own decision vector per subband. For example, for 2 levels this effectively brings the number of subbands to $q \cdot 21 = 63$, where q is the number of feature sets. Note that although we have decreased the number of features significantly, we have also increased the number of classifiers, because now we have one classifier per subband. Evaluating this computational trade-off is a task for future work.

Feature sets we considered are those originally used in [15] (without wavelet and Gabor features, which are MR): texture (T_1), morphological (M) and Zernicke moments (Z).

Texture Features. In our previous work [11] [16], we found that texture features are the most discriminative. We modified the standard Haralick texture features we call T_1 [17] into a new set T_3, by separating vertical/horizontal from diagonal features, as follows:

$$f_i^{(T_3)} = \frac{f_{H,i} + f_{V,i}}{2}, \qquad f_{i+13}^{(T_3)} = \frac{f_{LD,i} + f_{RD,i}}{2}, \quad (1)$$

where f_i are the original Haralick texture features in horizontal (H), vertical (V) and diagonal directions (LD, RD) and $i = 1, \ldots, 13$, yielding a total of 26 features in the set T_3.

Weighting Algorithm. The weighting part combines all of the subband decisions into one. We use weights for each subband to adjust the impact that a particular subband has on the overall decision made by the classification system. If the weights are chosen such that the no decomposition weight is equal to 1, and all other weights are 0, we will achieve the same output vector as we would have without using the adaptive MR system. Therefore, we know that there is a weight combination that will do at least as well as the generic classifier (when no MR is involved) in the training phase. Our goal is to decide how to find the weight vector that achieves the highest overall classification accuracy on the data set.

We developed two versions of the weighting algorithm: open-form and closed-form. The difference between the open and closed-form algorithms is that in the open-form version we optimize classification accuracy on the training set as opposed to the closed-form where we look for the least-squares solution.

The classification block in Figure 1 consists of a neural network. It outputs a series of decision vectors for each subband of each training image. Each decision vector D_s contains 3 numbers (because we have 3 classes) that correspond to the "local" decisions made by subband s for a specific image.

If using the open-form algorithm, we initialize all the weights, and a global decision vector is computed using a weighted sum of the local decisions. An initial class label will be given to an image using this global decision vector. If that class label is correct, we go to the next image. If it is incorrect, we look at the local decisions of each subband and adjust the weights of each subband as follows:

$$w_s^{iter} = \begin{cases} w_s^{iter-1} \cdot (1+\epsilon) & \text{if subband } s \text{ is correct,} \\ w_s^{iter-1} \cdot (1-\epsilon) & \text{otherwise,} \end{cases}$$

where $iter$ is the iteration number and ϵ is a small positive constant. This can be viewed as a reward/punishment method where the subbands taking the correct decisions will have their weights increased, while those taking wrong decisions will have their weights decreased. We continue cycling through the images until there is no increase in classification accuracy on the training set for a given number of iterations.

The closed-form solution does not use an iterative algorithm. Instead, it concatenates all the decision vectors of all the images into a matrix Γ where each column represents one image. We can then solve for the vector of weights W using least squares in the following system:

$$\Gamma W = T \quad (2)$$

where Γ is of size $3R \times S$, R is the number of training images, S is the number of subbands, and the vector T is the target vector.

Screening. For each time-lapse series, we consider slices at three time points; the first is during the time when Stage 1 is expected to occur, the second is during the time Stage 2 is expected to occur, and likewise for the third time point (these times are known). We then determine normal/delayed/abnormal tags by comparing the

Tagging Chart					
	(1,2,3)	(1,1,2)	(1,2,2)	(1,1,1)	(1,3,3)
Tag	normal	delayed	delayed	abnormal	abnormal

Table 1. Tagging chart. All combinations starting with 2 or 3 will be assumed to be a classifier mistake. Those combinations should be converted to (1,x,y) where x and y are the original stage determination. Any combination starting with 1 and not in the above chart is assumed to be abnormal.

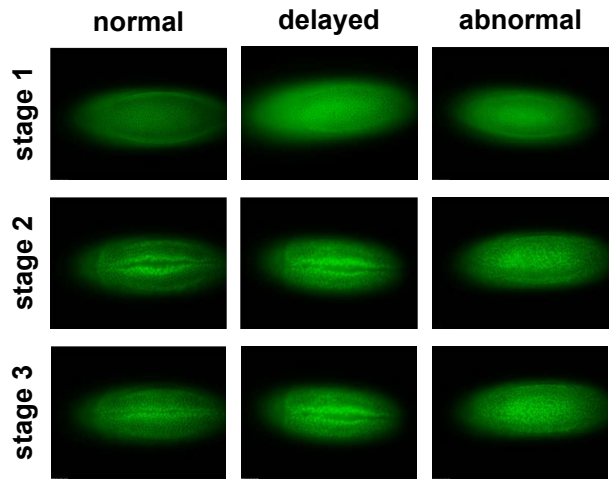

Fig. 2. Representative examples of each stage. Top: Stage 1, no ventral furrow, for normal (t=30min), delayed (t=60min) and abnormal (t=20min) embryos. Middle: Stage 2, ventral furrow start, for normal (t=60min), delayed (t=110min) and abnormal (t=72min) embryos. Bottom: Stage 3, ventral furrow closed, for normal (t=75min), delayed (t=140min) and abnormal (t=82min) embryos.

expected stages with what the classifier outputs for each set of time points. For example, if the three time points are classified as (1,2,3) (numbers refer to stages), then this is a normal image series. If the classifier labels the images as (1,1,2), then this is a delayed image series. If the classification is (1,1,1), then this is abnormal because it means development did not occur at all. For each combination, we assign a normal/delayed/abnormal tag. Our current assignment is given in Table 1. Of course, it is possible that the sequence (1,1,1) is the correct classification in the first case and incorrect in the last two, leading to an incorrect tag. We will assume that any combination starting with 2 or 3, that is, (2,x,y) or (3,x,y), is a classifier mistake and will convert it into (1,x,y). The combinations starting with 1 not shown in Table 1 are assumed to lead to abnormal tags.

3. EXPERIMENTAL RESULTS

3.1. Data Set

The data set consists of 60 time-lapse z-stacks (3D volumes in time). The stacks are acquired roughly every 10 minutes. The number of slices per stack varies; it is 5 slices for normal sets and 7 slices for delayed/abnormal. The number of time points is typically 15 for normal/abnormal and around 30 for delayed. All the slices have been tagged by a human expert so we have reliable ground truth.

3.2. Experimental Setup and Results

Base System (NMR). We denote the base system without the multiresolution block as *no MR (NMR)*. We used a two-layer neural network classifier. The first layer contains a node for each of the input features, each node using the Tan-Sigmoid transfer function. The second layer contains a node for each output and uses a linear transfer function. With this layout, there is an input for each feature and an output for each class. We then train the neural network using a one-hot design. Since each output from the second layer corresponds to a class, when training, each training image will have an output of 1 for the class of which it is a member and a 0 for all other classes. To maximize the use of our data, our training process of the neural network block uses five-fold cross validation. We train for 25 epochs, that is, the entire training set is presented 25 times to the neural network.

We ran the classifier on the combinations of T_1, T_3, M feature sets. We did not use the set Z as in previous experiments it did not show great promise, and is expensive to compute. The results are given in the first row of Table 2. As expected, T_3 outperforms T_1. However, adding the set M did not yield an improvement in accuracy, and even decreased it.

MR Basis Classification (MRB). We now implement our main idea of adding an MR block in front of feature computation and classification, as in Figure 1. We start with the MR decomposition being a basis expansion, and thus, we term this *MR basis classification (MRB)*. We grow a full MR tree with 2 levels. The classification system uses all the subspaces from the root (the original image) to the leaves of the tree. Hence, the total number of subbands used is 21 ($1+4+4^2$). We used the simplest, Haar filters in the decomposition, where the lowpass is given by $g = (1, 1)/\sqrt{2}$ whereas the highpass is $h = (1, -1)/\sqrt{2}$. This is done first in the horizontal direction and then in the vertical one, producing 16 outputs (subbands) on the second level of the decomposition. There are many other MRB blocks possible, the investigation of other ones is left for future work.

We grow a full tree to two levels with Haar filters. We then test the system with T_1, T_3 and M combinations of feature sets, a neural network classifier and the weighting algorithm (open form (OF) or closed form (CF)). The classifier is evaluated using nested cross validations (five-fold cross validation in the neural networks block and ten-fold during the weighting process). One problem with this technique is that the initial ordering of the images determines which images are grouped together for training and testing in each fold of the cross validation. A different original ordering of the images would result in different groupings, and thus a different overall result. We solve this problem by running multiple trials, each with a random initial ordering of the images. The mean result of these trials is taken as our true classification accuracy.

The second and third rows of Table 2 show the results of this experiment. The second row gives the results for the open-form weighting algorithm (OF), while the third row gives the same results the closed-form one (CF). For the OF (second row), we see that by adding MR, we achieve significant improvement on the NMR (no multiresolution) in all cases, achieving the best accuracy of 89.89% for the (T_3, M) combination. For the CF (third row), we conclude that the CF outperforms OF in each case, achieving the best accuracy of 92.78% for the (T_3, M) combination.

MR Frame Classification (MRF). In our work with images of subcellular protein location, we found that adding redundancy in the

Classification Accuracy [%]				
2D	Weight	T_1	T_3	T_3, M
NMR	NW	82.94	88.39	78.33
MRB	OF	88.11	91.06	89.89
	CF	90.94	92.22	92.78
MRF	OF	83.44	89.95	90.51
	CF	84.83	91.06	**93.17**
3D	Majority rule on 2D			**98.35**

Table 2. Classification accuracy for 2D slices. We use these in majority voting classification for 3D stacks yielding the accuracy of **98.35**%. NMR stands for no multiresolution, MRB for MR bases and MRF for MR frames. OF denotes open-form weighting algorithm while CF denotes closed-form weighting algorithm. NW denotes no weighting as there is no MR block in front.

MR transform helped, that is, when we used frames instead of bases, accuracy increased. We postulated this was the case because the MRB were shift variant. Here, the results with frames (à trous MR block), are given in the fourth and fifth rows of the table, for the OF and the CF weighting algorithms, respectively. Interestingly enough, for T_1 and T_3 alone, frames did not improve the accuracy. However, for the best-performing combination from before, (T_3, M), frames improve the performance, reaching a high accuracy of 93.17%.

3D Classification. The above classification system classifies 2D slices. Since we have access to 3D stacks, we make use of those, by classifying three slices out of each stack and then making a decision using a majority rule. The classification results for adjacent slices were 92.38% and 91.64% using (T_3, M) features and closed-form weighting. Using the majority rule process, the classification accuracy reaches 98.35%. We can use the same process in time to improve the screening, but since we do not have enough slices in time and need to acquire time-lapse series with better time resolution, this is left for future work. Note that using the majority rule assumes the slices in a 3D stack to be independent. We have not verified this assumption and will leave it for future work.

4. CONCLUSIONS AND FURTHER WORK

We presented an algorithm for automated screening of *Drosophila* embryos based on ventral furrow formation. The algorithm achieves an accuracy of 98.35% with a potential for even higher numbers once higher time-resolution time-lapse series are acquired. We build this as part of an image analysis/processing toolbox for high-throughput Drosophila embryo RNAi screens. Future work involves both improving the present algorithm as well as adding toolbox functionalities by developing algorithms for other problems within the screen (acquisition, segmentation, etc.).

5. REFERENCES

[1] G. M. Rubin, M. D. Yandell, J. R. Wortman, G. L Gabor Miklos, C. R. Nelson, I. K. Hariharan, M. E. Fortini, P. W. Li, R. Apweiler, W. Fleischmann, and et al., "Comparative genomics of the eukaryotes," *Science*, vol. 287, pp. 2204–2215, 2000.

[2] M. D. Adams, S. E. Celniker, R. A. Holt, C. A. Evans, J. D. Gocayne, P. G. Amanatides, S. E. Scherer, P. W. Li, R. A. Hoskins, R. F. Galle, and et al., "The genome sequence of Drosophila melanogaster," *Science*, vol. 287, pp. 2185–2195, 2000.

[3] S. Richards, Y. Liu, B. R. Bettencourt, P. Hradecky, S. Letovsky, R. Nielsen, K. Thornton, M. J. Hubisz, R. Chen, R. P. Meisel, and et al., "Comparative genome sequencing of Drosophila pseudoobscura: Chromosomal, gene, and cis-element evolution," *Genome Research*, vol. 15, pp. 1–15, 2005.

[4] A. E. Carpenter and D. M. Sabatini, "Systematic genome-wide screens of gene function," *Nat. Rev. Genet.*, vol. 5, pp. 11–22, 2004.

[5] Z. Zhou and H. Peng, "Automatic annotation and recognition of gene expression patterns of fly embryos," *BMC Bioinformatics*, 2007.

[6] A. Fire, S. Xu, M. K. Montgomery, S. A. Kostas, S. E. Driver, and C. C. Mello, "Potent and specific genetic interference by double-stranded rna in caenorhabditis elegans," *Nature*, vol. 391, pp. 806–811, 1998.

[7] S. Zappe, M. Fish, M. P. Scott, and O. Solgaard, "Automated MEMS-based Drosophila embryo injection system for high-throughput RNAi screens," *Lab Chip*, vol. 6, pp. 1012–1018, 2006.

[8] E. Yeh, K. Gustafson, and G. L. Boulianne, "Green fluorescent protein as a vital marker and reporter of gene expression in Drosophila," *Proc. Nat. Academy Sci.*, vol. 92, pp. 7036–7040, 1995.

[9] L. Gong, M. Puri, M. Ünlü, M. Young, K. Robertson, S. Viswanathan, A. Krishnaswamy, S. R. Dowd, and J. S. Minden, "Drosophila ventral furrow morphogenesis: A proteomic analysis," *Development*, vol. 131, no. 3, pp. 643–656, 2004.

[10] N. Saito and R. Coifman, "Local discriminant bases and their applications," *Math. Imaging Vision*, vol. 5, pp. 337–358, 1995.

[11] T. Merryman, K. Williams, G. Srinivasa, A. Chebira, and J. Kovačević, "A multiresolution enhancement to generic classifiers of subcellular protein location images," in *Proc. IEEE Int. Symp. Biomed. Imaging*, Arlington, VA, Apr. 2006, pp. 570–573.

[12] R. R. Coifman, Y. Meyer, S. Quake, and M. V. Wickerhauser, "Signal processing and compression with wavelet packets," Tech. Rep., Yale University, 1991.

[13] N. Saito and R. R. Coifman, "Local discriminant bases," in *Proc. SPIE Conf. Vis. Commun. and Image Proc.*, 1994, pp. 2–14.

[14] P. Hennings Yeomans, J. Thornton, J. Kovačević, and B. V. K. V. Kumar, "Wavelet packet correlation methods in biometrics," *Appl. Opt., sp. iss. Biometric Recognition Systems*, vol. 44, no. 5, pp. 637–646, Feb. 2005.

[15] K. Huang and R. F. Murphy, "From quantitative microscopy to automated image understanding," *Journ. Biomed. Optics*, vol. 9, pp. 893–912, 2004.

[16] A. Chebira, Y. Barbotin, C. Jackson, T. Merryman, G. Srinivasa, R. F. Murphy, and J. Kovačević, "A multiresolution approach to automated classification of protein subcellular location images," *BMC Bioinformatics*, 2007, Submitted.

[17] R. M. Haralick, "Statistical and structural approaches to texture," *Proc. IEEE*, vol. 67, pp. 786–804, 1979.