Automatic localization of interest points in zebrafish images with tree-based methods

Olivier Stern[1], Raphaël Marée[1,2], Jessica Aceto[3], Nathalie Jeanray[3],
Marc Muller[3], Louis Wehenkel[1], and Pierre Geurts[1]

[1] GIGA-Systems Biology and Chemical Biology, Dept. of EE and CS
University of Liège, Belgium
[2] GIGA Bioinformatics Core Facility, University of Liège, Belgium
[3] GIGA-Development, Stem Cells and Regenerative Medicine,
Molecular Biology and Genetic Engineering
University of Liège, Belgium

Abstract. In many biological studies, scientists assess effects of experimental conditions by visual inspection of microscopy images. They are able to observe whether a protein is expressed or not, if cells are going through normal cell cycles, how organisms evolve in different experimental conditions, etc. But, with the large number of images acquired in high-throughput experiments, this manual inspection becomes lengthy, tedious and error-prone. In this paper, we propose to automatically detect specific *interest points* in microscopy images using machine learning methods with the aim of performing automatic morphometric measurements in the context of Zebrafish studies. We systematically evaluate variants of ensembles of classification and regression trees on four datasets corresponding to different imaging modalities and experimental conditions. Our results show that all variants are effective, with a slight advantage for multiple output methods, which are more robust to parameter choices.

1 Context, motivation, and strategy

The zebrafish is a well-known model organism increasingly used for biological studies on development, gene function, toxicology, and pharmacology. In addition to its major biological advantages (ease of reproduction, quick growth, genome close to the human's), the fact that embryos are transparent eases microscopic observations. More specifically, in bone research studies, the skeleton can be observed at different stages of development combined with appropriate staining [1, 13]. From these images, one seeks to perform morphometric measurements of the cartilage skeleton to describe the effects of different experimental conditions such as chemical treatments or gene knock-downs. Interesting measurements include the length of cartilages or angles defined by specific interest points.

Traditionally, effects of biological experiments on zebrafish embryos are evaluated manually through microscopic observation. However, due to the large number of experimental protocols, chemical substances, acquisition modalities, and

This is the author's version of the work. The definitive version will be published in Proc. 6th IAPR International Conference on Pattern Recognition in Bioinformatics (2011), Springer LNCS.

the recent availability of high-throughput imaging equipments, visual inspection of zebrafish images by experts is becoming a limiting factor in terms of time and cost. Moreover, for humans it is often hard to distinguish visually subtle changes and in particular to perform measurements in a reproducible way. Using traditional low-level image processing methods (e.g. those based on thresholding and mathematical morphology) would also be limiting because a significant number of factors make images quite different from an experiment to another hence it would require tuning image processing operations for each and every experiment. Indeed, factors such as biological preparation protocols, imaging acquisition procedures, and experimental conditions produce very different types of images, such as those considered in this paper and illustrated by Figure 1.

These observations motivated us to consider generic machine learning methods to speed-up the reproducible extraction of quantitative information from these images. In our approach, experts first encode manually the localization of interest points within a few training images, for each batch of images. These annotations are then used to train either classification or regression models that are used in order to locate in a fully automatic way these interest points in the remaining images of the current batch.

2 Methods

As stated before, we follow a supervised learning approach, ie. we exploit manually annotated images (see Figure 1 for a few examples) where interest points coordinates have been localized by experts to train models able to predict those interest points in new, unseen images. The approach first extracts subwindows (or patches) around points of interest and at other randomly chosen positions within images, describe these patches by various visual features, and then either a classification or a regression model is built. In the classification scheme, the model is trained to predict whether the central pixel of a subwindow is an interest point or not (a binary classification problem). In the regression scheme, the model predicts the distance between the central pixel of a subwindow and the interest point. These models are built using either single output (one model per interest point) or multiple outputs (one model predicts simultanously all the interest points).

Table 1 describes the overall algorithmic approach within the single output setting. The different steps of this procedure are further explained in the following subsections.

2.1 Extraction and description of subwindows

The input of our learning algorithms is a learning set of subwindows of size l x l extracted within the training images in the following way: (i) for each pixel located within a circular region of radius r around an interest point a subwindow centered on this pixel is extracted; (ii) a certain number of subwindows are randomly extracted from the rest of the image. In our experiments we always

Table 1. Training and testing algorithms (single output setting)

Parameters: radius of the interest region r, subwindows size l, *method*: either 'classification' or 'regression', a subwindow feature extractors $x(.;l)$, Extra-trees parameters T and K.

Train(LS)
Input: a learning sample of N images with the interest point position:

$$LS = \{\langle I_i, (p_{x,i}, p_{y,i})\rangle | i = 1, \ldots, N\},$$

Output: an ensemble of trees defined on subwindows features
- $LS_{sw} = \emptyset$
- For each pair $\langle I, (p_x, p_y)\rangle \in LS$:
  - $S_p = \emptyset$
  - add in S_p all positions (p'_x, p'_y) such that $(p'_x - p_x)^2 + (p'_y - p_y)^2 < r^2$.
  - Let $P = |S_p|$ the size of S_p. Add in S_p $2P$ positions (p'_x, p'_y) randomly drawn in the image such that $(p'_x, p'_y) \notin S_p$
  - For each (p_x, p_y) in S_p, add $\langle x(p_x, p_y; l), y\rangle$ in LS_{sw} where:
    - $x(I, p_x, p_y; l) \in I\!R^m$ is the feature descriptors (of size m) of the $l \times l$ subwindow centered at (p_x, p_y) in I
    - y is either $1((p'_x - p_x)^2 + (p'_y - p_y)^2 < r^2)$ if *method*='classification' or $(p'_x - p_x)^2 + (p'_y - p_y)^2$ if *method*='regression'.
- Return the ensemble of trees obtained by the Extra-trees algorithm applied on LS_{sw}

Predict(I, ens)
Input: an image I, an ensemble of trees ens returned by the function **Train**
Output: a prediction $(\hat{p}_x, \hat{p}_y)$ of the position of the interest point in I
- For all pixels (p_x, p_y) in I, compute the predictions $\hat{y}(p_x, p_y)$, by propagating the feature vectors $x(I, p_x, p_y; l)$ into ens.
(*NB: $\hat{y}(p_x, p_y)$ is the predicted probability of class j in classification and the predicted distance to the interest point in regression.*)
- Compute as a prediction for the position of the interest point:

$$(\hat{p}^j_x, \hat{p}^j_y) = median(\{(p_x, p_y) \in I | \hat{y}^j(p_x, p_y) = \bar{y}\},$$

where:

- $\bar{y} = \arg\max_{(p_x, p_y)} \hat{y}(p_x, p_y)$ if *method*='classification',
- $\bar{y} = \arg\min_{(p_x, p_y)} \hat{y}(p_x, p_y)$ if *method*='regression'.

---

used twice as many "other" subwindows as the number of interest point specific ones, which increases proportionally to the radius r.

Input features. Each subwindow is then processed to compute the input features using three visual cues:

– Color and grayscale: Each pixel of a subwindow is decomposed in the Red-Green-Blue (RGB) color space (3 features per pixel), in Hue-Saturation-Value (HSV) color space (3 features per pixel), and in grayscale (luminance, 1 feature per pixel).
– Edges: The gradient of the Sobel operator is applied on each pixel and its direct neighbours. Considering A as a 3×3 matrix of a pixel and its eight neighbours in grayscale, we define the gradient G (1 feature per pixel) as:

$$G = \sqrt{G_x^2 + G_y^2} \quad \text{with } G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \times A \text{ and } G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \times A,$$

where × denotes the scalar product.
– Texture: We use the basic version of the local binary pattern (LBP) [16] to describe the texture of subwindows. Each pixel of a subwindow is compared to its 8 neighbors. If the intensity of the center pixel is greater than the compared neighbor, we encode it by 1, and otherwise by 0. This gives an 8-digit binary number per pixel, that we convert in decimal. We compute the histogram of these numbers over the subwindow, yielding a 256-dimensional feature vector.

Overall, we thus use $m = (3 + 3 + 1 + 1) \times l \times l + 256$ numerical features to describe the visual content of the subwindows.

Outputs. In the single output classification approach, the output is binary and equal to 1 if the central pixel is an interest point, 0 otherwise. In the multiple-output approach, the output is a vector of $N_p + 1$ class-indicator variables, where the N_p first components correspond to the N_p types of interest points whereas the last component corresponds to the background.

In the single output regression approach, the output is a number reflecting the distance of the center pixel to the interest point. In the multiple output approach, the output is a vector of N_p numbers corresponding to the distances to the N_p interest points.

2.2 Model construction using extremely randomized tree ensembles

Starting with the learning set of subwindows at the top-node, the Extra-Trees algorithm [10] builds an ensemble of T fully developed decision trees. At each node, it generates tests on input variables (features) in order to progressively partition the input space into hyper-rectangular regions where the output is constant. In order to select relevant tests, k features are chosen at random at each node, where the filtering parameter k can take all possible values from 1 to the total number m of features. For each of these k features, a numerical value is randomly chosen within the range of variation of that feature in the subset of subwindows available in the current tree node. The score of each binary test is then computed on the current subwindow subset, and the best test among the k tests is chosen to split the current node into two child nodes. For single output classification and regression trees, we use CART's standard score measures, i.e., Gini index reduction in classification and output variance reduction in the case of regression [5]. In the case of multiple classification or regression outputs, prediction at leaf nodes of the trees are extended to be vectorial and we use as a score measure the sum of the scores for each individual output (see e.g. [4, 7] for a treatment of multiple output trees).

2.3 Prediction in test images

To localize interest points in a new, unseen image, we extract a subwindow centered at every pixel position and propagate it into the trees to predict one or more value(s) according to the model used. Indeed, as explained in Section 2, the output of classification and regression models will be different.

In the classification scheme, a model outputs probability estimates to determine if the central pixel of a subwindow is or not an interest point. The multiple output approach will consider all the interest points at once, while in the single output approach each model is applied separately to predict probability estimates for each interest point. In order to produce the coordinates of one interest point within a test image, we then compute the median of all pixel positions which obtained the highest predicted probability estimate.

In the regression scheme, we predict the euclidian distance from the central pixel of a subwindow to our interest points. In the multiple output variant, we consider as ouputs the distances to all the interest points at once, while in the single output scheme each model predicts the distance to a specific interest point. To obtain the predicted coordinates of one particular interest point within a test image, we compute the median of all pixel positions which obtained the smallest predicted distance.

2.4 Related work

Various studies use computational techniques for zebrafish image quantification but none addresses the problem of skeleton/cartilage morphometric measurements using machine learning methods, to the best of our knowledge. In [2], a study of embryo images submitted to toxicological treatments is proposed. They aim at observing the mortality rate depending on the toxicological concentrations, by extracting image features (e.g. variance of pixel values) to distinguish dead and alive embryos. Images are then labeled by experts and classified thanks to the Matlab Gait-CAD toolbox in only two classes. [3] describes a way to automatically obtain images of zebrafish embryos thanks to a motorized microscope, and classifies them manually into phenotypes. [19] developed an automatic system of data acquisition and embryos' analysis in multi-well plates, where manual intervention is still needed to produce analysis routines based on several image segmentations. More recently, another approach to classify embryos of zebrafish depending on several phenotypes has been proposed in [12]. Manually acquired images of zebrafish embryos are first pre-processed to standardize images which are then submitted to a phenotypic classification. The supervised learning algorithm used is based on random subwindows extraction in images [15], their description by raw pixel values, and the use of ensembles of extremely randomized trees [10] to classify these subwindows hence images.

Beyond studies involving Zebrafish images, one can see similarities between our work and some applications in the broader pattern recognition literature. In the field of face recognition, multi-stage classification and regression approaches have been proposed (e.g. [6, 8]) to localize facial features (e.g. eyes) as a preliminary step for face recognition. In scene and object recognition tasks, [18] emulates two generic interest point detectors (Hessian-Laplace and Kadir-Brady) using boosting approaches. [9, 14] use randomized trees for object detection and real-time tracking. In medical imaging, [17] uses regression trees to detect bounding boxes of organs in CT images.

In protein bioinformatics, [11] compares classification and regression approaches for protein binding site prediction using patch based predictors similarly as our visual interest point localization with subwindow predictors.

3 Experiments

3.1 Datasets

We apply our method on four different image datasets illustrated in Figure 1. The image size in all datasets is 400 pixels × 300 pixels.

- CTL: a batch of 15 wild type zebrafish images stained with alcian blue. The four interest points are located on the skeleton.
- DRUG: a second batch of 20 zebrafish images from a toxicology experiment (also stained with alcian blue but with slightly different imaging settings). From the biological point of view, the goal is to compare these to the CTL database so as to quantify morphometric changes due to drug treatment.
- RED: a batch of 24 zebrafish control images stained with alizarin red to detect the bone skeleton.
- EMBRYO: a batch of 20 zebrafish embryo images where the four interest points characterize body parts.

3.2 Evaluation protocols

Considering the experts' annotations in all images for each database, we want to evaluate the performance of the classification and the regression methods for

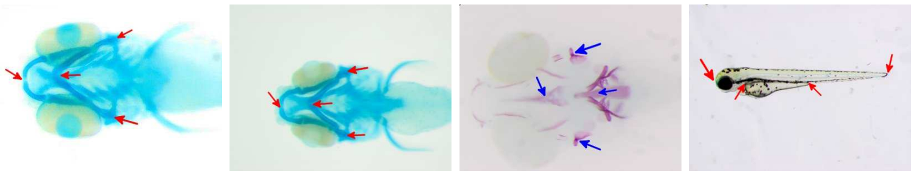

Fig. 1. An image from each database (CTL, Drug, Red, Embryo) where their manually annotated interest points are pointed by an arrow.

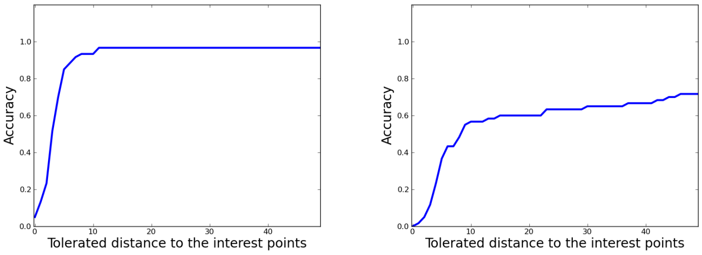

Fig. 2. Distance accuracy graphs for the CTL database (multiple-output classification setting; $T = 20, k = 10, r = 5$): left $l = 21$; right $l = 5$.

single and multiple outputs. To do so, we will perform a leave-one-out cross validation for different values of method parameters: the number of trees in the ensemble (T), the value of the filtering parameter (k), the radius of the circular region around the interest points (r), and the size of the subwindows (l). The default values of these parameters are $T = 20$, $k = 10$, $r = 5$, and $l = 21$, leading to a good compromize between accuracy and computational complexity.

For each experiment (i.e. each leave-one-out computation on one dataset and with a given method setting and parameter values), we have constructed a distance accuracy graph which represents the percentage of predicted points within a distance d to the interest points. Figure 2 shows two such accuracy graphs for one of the variants, illustrating the effect of the parameter l. The faster the curves reach the upper bound of 100%, the better the approach. Hence, we will use the area under these curves (AUC) to assess and compare the different settings and parameter values.

For our four datasets, the results of our empirical assessment are in Figures 5, 6, 7, and 8. Each graph shows the AUC of the four methods as a function of one of the four parameters, the other parameters being kept constant and set to their default values. Notice that overall, these experiments involved roughly 2500 computer jobs (each one corresponding to one experiment).

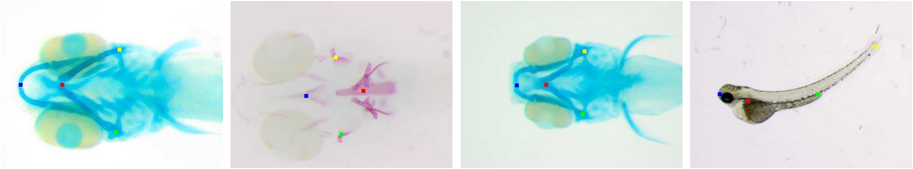

Fig. 3. Prediction of interest points in each database: multiple output regression approach with default parameter settings ($T = 20, k = 10, r = 5, l = 21$).

3.3 Results and observations

At first, we can see on the prediction example in Figure 3 that the results are visually very satisfying. We observed such results in each database for each method setting, provided that the parameter values are well chosen.

Let us further investigate the results according to the different AUC produced as a function of the parameters and method settings. According to figures 5, 6, 7 and 8 we can make the following observations:

- The parameter k has little impact on the accuracy
- For higher values of r, the single output regression method obtains the best results
- The size of subwindows is very important, whatever the method. Only high values of l obtain good performances
- At low values of l, T and r, multiple output methods obtain generally better results than single output methods
- At the light of these graphs, it's not obvious that there is a best method to resolve our problem, but the multiple output settings appear to be more robust with respect to parameter values.

A last remark concerns the difference between the output of the classification and the regression methods. Figure 4 illustrates the fact that the classification setting often finds several 'best predictions' while in the regression setting we observed that in general only one position is predicted as the most likely one.

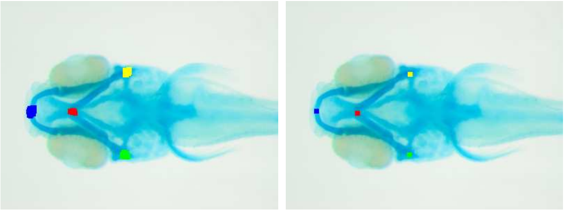

Fig. 4. Output of the best predictions in classification (left) and regression (right) methods for a same image, before evaluating the median.

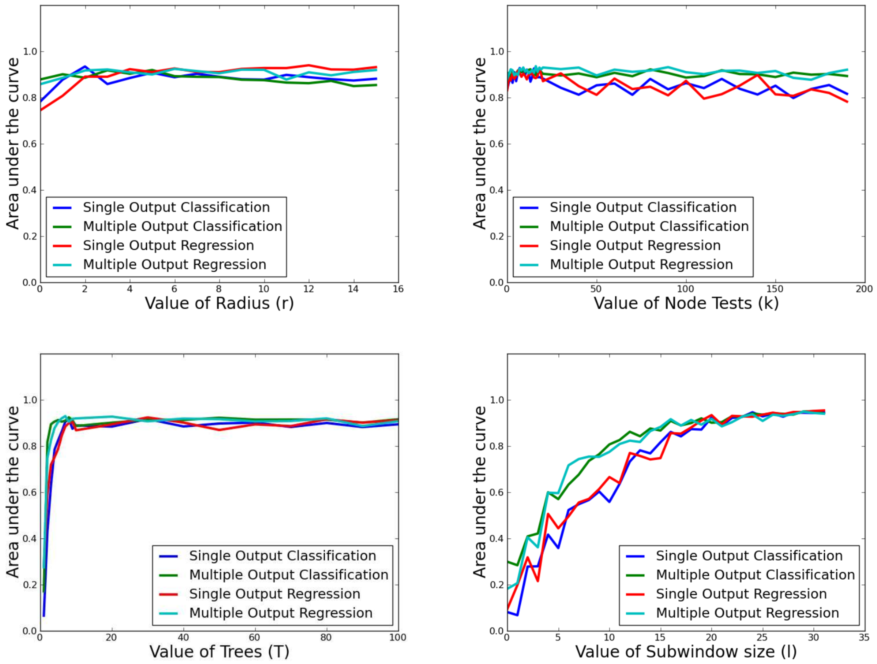

Fig. 5. [CTL database] Area under the curve computed from the distance accuracy graphs for the different values of the parameters r, k, T, l.

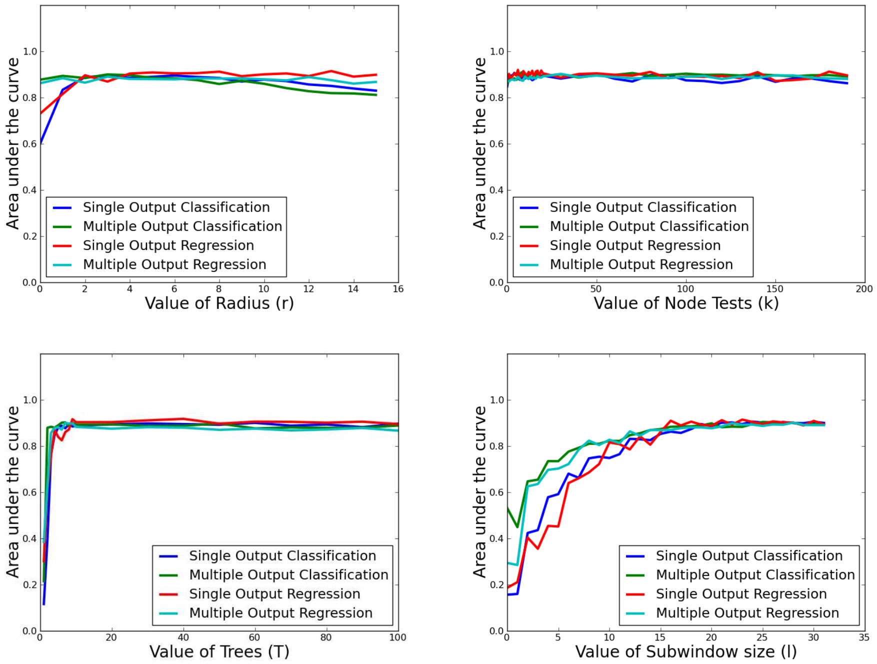

Fig. 6. [DRUG database] Area under the curve computed from the distance accuracy graphs for the different values of the parameters r, k, T, l.

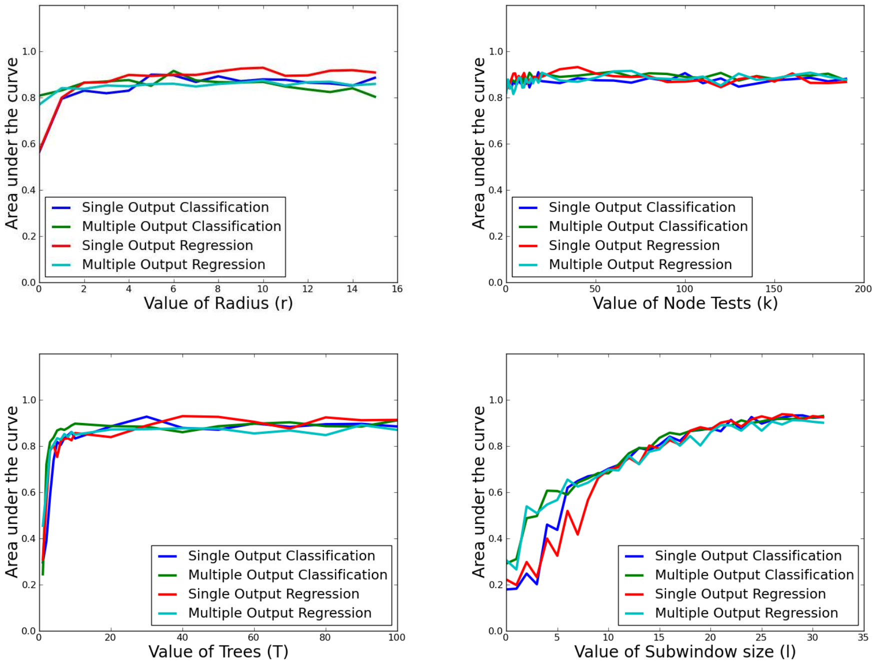

Fig. 7. [RED database] Area under the curve computed from the distance accuracy graphs for the different values of the parameters r, k, T, l.

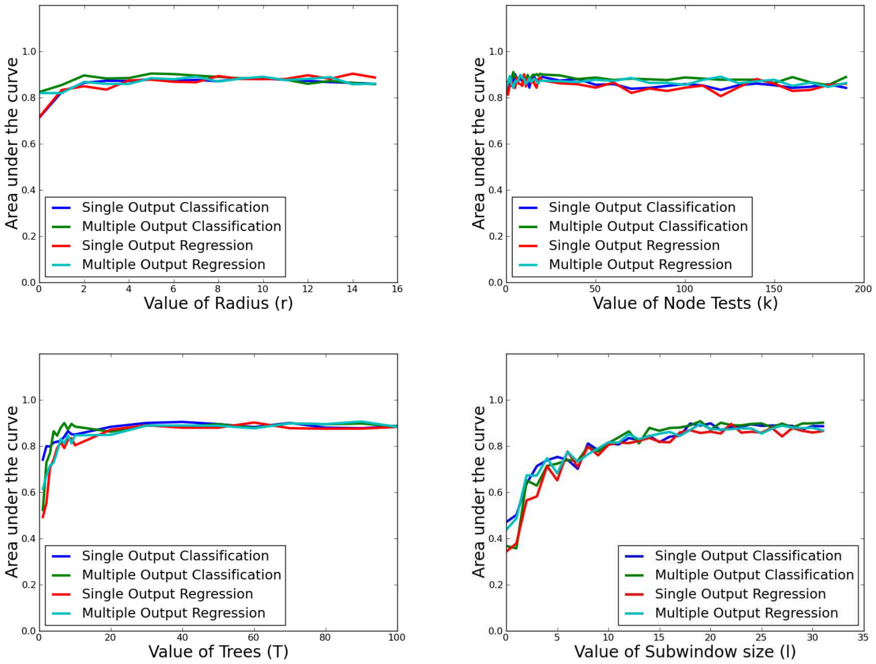

Fig. 8. [EMBRYO database] Area under the curve computed from the distance accuracy graphs for the different values of the parameters r, k, T, l.

4 Conclusion and future work

In this paper, we tackled the task of specific interest point detection in zebrafish images using machine learning methods based on ensembles of randomized regression and classification trees. We compared different settings (multiple vs single output, regression vs classification) on four imaging datasets and have studied the effect of various parameters on accuracy.

Our study shows that all approaches give good results provided that parameters are well chosen. We also found that the parameter which has the strongest influence is the window size, and that the multiple output setting is less sensitive to parameter choices than the single output setting.

Although this work did not focus on the computational aspects, we notice that training and prediction with tree-based ensemble methods is highly scalable with respect to dataset size and feature space dimensionality. As a matter of fact, the main computational burden of the approach is related to the extraction of subwindows from the original images and the subsequent feature computations. These are specially demanding at the stage of prediction, since one subwindow for each pixel has to be extracted, represented and then classified.

Future work will thus look at computational optimizations and further algorithmic optimizations and more extensive large scale validation studies, specially in the context of toxicology studies. In terms of accuracy, we note that in this paper we used the precision of interest point localization as a criterion, while in practical biological studies these are used in order to compute more complex geometrical features or phenotype classifications, generally involving the computation of several interest points. Hence, accuracy evaluations should also be made on these end-outcomes used by biologists to assess statistical significance of the impact of the considered experimental conditions.

Acknowledgments

This paper presents research results of the ARC BIOMOD, the Interuniversity Attraction Poles Programme (IAP P6/25 BIOMAGNET), initiated by the Belgian State, Science Policy Office, and by the European Network of Excellence, PASCAL2. RM is supported by the GIGA interdisciplinary cluster of Genoproteomics of the University of Liège with the help of the Walloon Region and the European Regional Development fund, and by the CYTOMINE research grant n°1017072 of the Walloon Region (DGO6). PG is a research associate of the FNRS, Belgium.

References

1. K. F. Adkins. Alizarin red s as an intravital fluorochrome in mineralizing tissues. *Stain Technol.*, 40:69–70, 1965.
2. R. Alshut, J. Legradi, U. Liebel, L. Yang, J. Van Wezel, U. Strähle, R. Mikut, and M. Reischl. Methods for automated high-throughput toxicity testing using zebrafish embryos. *R. Dillmann et al.*, pages 219–226, 2010.

3. D. Arslanova, T. Yang, X. Xu, S. Wong, C. Augelli-Szafran, and W. Xia. Phenotypic analysis of images of zebrafish treated with alzheimer's γ-secretase inhibitors. *BMC Biotechnology*, pages 10–24, 2010.
4. H. Blockeel, L. De Raedt, and J. Ramon. Top-down induction of clustering trees. In *Proceedings of ICML 1998*, pages 55–63, 1998.
5. L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees.* Wadsworth and Brooks, Monterey, CA, 1986.
6. P. Campadelli, R. Lanzarotti, and G. Lipori. Eye localization: a survey. *In The Fundamentals of Verbal and Non-verbal Communication and the Biometrical Issue*, 2007.
7. M. Dumont, R. Marée, L. Wehenkel, and P. Geurts. Fast multi-class image annotation with random subwindows and multiple output randomized trees. In *Proc. of the International Conference on Computer Vision Theory and Applications*, volume 2, 2009.
8. M. Everingham and A. Zisserman. Regression and classification approaches to eye localization in face images. In *Proc. of the 7th Int. Conf. on Automatic Face and Gesture Recognition*, pages 441–448, 2006.
9. J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky. Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1022–1029, 2011.
10. P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 36:3–42, 2006.
11. J. Giard, J. Ambroise, J.-L. Gala, and B. Macq. Regression applied to protein binding site prediction and comparison with classification. *BMC Bioinformatics 2009, 10:276*, 10(276), 2009.
12. N. Jeanray, R. Marée, B. Pruvot, O. Stern, P. Geurts, L. Wehenkel, and M. Muller. Phenotype classification of zebrafish embryos by supervised learning. Poster at Belgian Dutch Conference on Machine Learning (Benelearn), 2011.
13. C. B. Kimmel, C. T. Miller, G. Kruze, B. Ullmann, R. A. BreMiller, K. D. Larison, and H. C. Snyder. The shaping of pharyngeal cartilages during early development of the zebrafish. *Dev. Biol.*, 203:245–263, 1998.
14. V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:1465–1479, 2006.
15. R. Marée, P. Geurts, J. Piater, and L. Wehenkel. Random subwindows for robust image classification. In Cordelia Schmid, Stefano Soatto, and Carlo Tomasi, editors, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, volume 1, pages 34–40. IEEE, June 2005.
16. T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29:51–59, 1996.
17. S. D. Pathak, A. Criminisi, J. Shotton, S. White, D. Robertson, B. Sparks, I. Munasinghe, and K. Siddiqui. Validating automatic semantic annotation of anatomy in dicom ct images. In *Proceedings of the Medical Imaging 2011: Advanced PACS-based Imaging Informatics and Therapeutic Applications*, 2011.
18. J. Sochman and J. Matas. Learning fast emulators of binary decision processes. *International Journal of Computer Vision*, 83:149–163, 2009.
19. A. Vogt, A. Cholewinski, X. Shen, S. Nelson, J. Lazo, M. Tsang, and N. Hukriede. Automated image-based phenotypic analysis in zebrafish enbryos. *Developmental Dynamics*, pages 238:656–663, 2009.