Learning to Recognize Plankton*

Tong Luo, Kurt Kramer
Dmitry Goldgof, Lawrence O. Hall
Dept. of Computer Science & Engineering
University of South Florida
Tampa, FL 33620
{tluo2, kkramer, goldgof, hall}@csee.usf.edu

Scott Samson, Andrew Remsen
Thomas Hopkins
Dept. of Marine Science
University of South Florida
St. Petersburg, FL 33701
samson@seas.marine.usf.edu

Abstract – *We present a system to recognize underwater plankton images from the Shadow Image Particle Profiling Evaluation Recorder. As some images do not have clear contours, we developed several features that do not heavily depend on the contour information. A soft margin support vector machine (SVM) was used as the classifier. We developed a new way to assign probability after multi-class SVM classification. Our approach achieved approximately 90% accuracy on a collection of images with minimal noise. On another image set containing manually unidentifiable particles, it also provided promising results. Furthermore, our approach is more accurate on the two data sets than a C4.5 decision tree and a cascade correlation neural network at the 95% confidence level.*

Keywords: plankton recognition; support vector machine; learning; probabilistic output.

1 Introduction

Recently, the Shadow Image Particle Profiling Evaluation Recorder (SIPPER) was developed to continuously sample plankton and suspended particles in the ocean [13]. The SIPPER uses high-speed digital line-scan cameras to record images of plankton and other particles, thus avoiding the extensive post-processing necessary with analog video particle images. The large sampling aperture of the sensor combined with its high imaging resolution (50 μm per pixel), means that it is capable of collecting tens of thousands of plankton images an hour. This soon would overwhelm a scientist attempting to manually classify the images into recognizable plankton groups. Therefore, an automated plankton recognition system is necessary to solve the problem or at the very least to help with the classification.

Tang [15] developed a plankton recognition system to classify plankton images from video cameras. The moment invariants and Fourier descriptor features from contour images were extracted. Also, granulometric features from the gray-level images were computed. Finally, a learning vector quantization neural network was used to classify examples. Tang [15] achieved 92% classification accuracy on a medium-size data set.

The project ADIAC (Automatic Diatom Identification and Classification) has been ongoing in Europe since 1998. Different feature sets and classifiers have been experimented with to recognize separate species of diatom taken from photo-microscopes. Loke [8] and Ciobanu [3] studied some new contour features. Santos [14] extended the contour features to multi-scale Gabor features together with texture features. Wilkinson [17] applied morphological operators to help extract both contour and texture information. Fischer [6] summarized these features and used ensembles of decision trees to classify the combined feature set. Greater than 90% overall accuracy was achieved on the diatom images.

However, images from previous work are of relatively good quality or at least with clear contours. Therefore, complicated contour features and texture information can be extracted easily. The SIPPER images, on the other hand, present several difficulties:

1. Many SIPPER images do not have clear contours. Some are partially occluded. Therefore, we cannot depend mainly on contour information to recognize the plankton.

2. The SIPPER image gallery includes many unidentifiable particles as well as different types of plankton.

3. The SIPPER images in our experiments are binary, thus lacking enough texture information.

Not depending heavily on contour information, several special features were developed in our system, and a support vector machine (SVM) [16] was used to classify the feature vectors. We developed a new way to compute probabilistic outputs from a multi-class support vector machine.

This paper is organized as follows. Section 2 introduces the binary SIPPER images used in our experiments. In Section 3, we discuss the preprocessing of the images and the extraction of the features. Section 4 describes the support vector machine and the way we

*0-7803-7952-7/03/$17.00 © 2003 IEEE.

assign the probability in a multi-class support vector machine. Experimental results for our system are detailed in Section 5. Finally we summarize our work and propose some ideas for future work in Section 6.

2 Image gallery

Domain experts built the training data as follows. The SIPPER images were first converted to binary images by choosing a proper threshold. A morphological closing operation was used to analyze the connectivity and then segment the binary images. Next, an expert manually classified the images into recognizable plankton groups. The rest of the unrecognizable images were put into the unidentifiable-particle group. Figure 1 contains typical examples from SIPPER images.

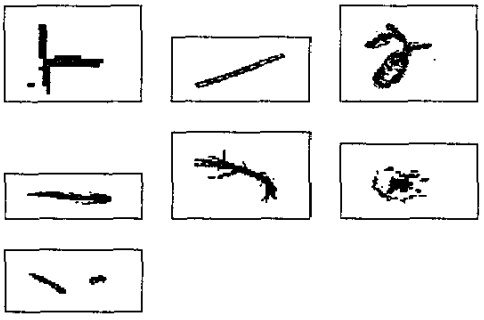

Figure 1: Figures from left to right, from top to bottom are typical examples of copepod, diatom, doliolid, larvacean, trichodesmium, protoctista and a manually unidentifiable particle.

3 Feature computation

The SIPPER images have a lot of noise around or on their bodies and some do not even have a clear contour, thus making contour features (Fourier descriptor [18] etc.) unstable and inaccurate. To solve this problem, we first preprocessed the images to suppress the noise. We only applied invariant moments and granulometric features, which are relatively stable with respect to noise and do not depend heavily on the contour image. To capture the specific information from our SIPPER image set, domain knowledge was used to extract some specific features such as size, convex ratio, transparency ratio, etc.

3.1 Noise suppression

We applied connected component analysis to eliminate the noise pixels far from the plankton bodies. In addition, a morphological closing operation was used to separate the holes inside the plankton body from the background [10].

3.2 Moment invariants

Moment features are widely used as general features in shape recognition. The standard central moments are computed as follows:

$(\bar{x}, \bar{y})$ is the center of the foreground pixels in the image. The $(p+q)$-order central moments are computed with every foreground pixel at (x, y):

$$\mu(p,q) = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q \qquad (1)$$

Then central moments are normalized by size:

$$\eta(p,q) = \frac{\mu(p,q)}{\mu(0,0)^{\frac{p+q}{2}}} \qquad (2)$$

Hu [7] introduced a way to compute the seven lower order moment invariants based on several nonlinear combinations of the central moments. Using the normalized central moments, we got the scale, rotation and translation invariant features. We computed the same 7 moment invariants on the whole object and the contour image after a morphological closing operation, respectively.

3.3 Granulometric features

Since the Hu moments only contain low order information from the image, we extracted several granulometric features [9] to capture the high order information. Granulometric features were computed by doing a series of morphological openings with different sizes of structure elements. Then we recorded the differences in size between the plankton with and without openings. Granulometric features are relatively robust to noise and have the inherent information of shape distribution. Tang [15] found that granulometric features were the most important features in his experiment.

We applied 3×3, 5×5, 7×7 and 9×9 squares as structure elements and did a series of morphological openings. Then differences in size were normalized by the original plankton size to obtain the granulometric features. Also, we applied 3×3, 5×5 and 7×7 squares as structure elements, and did a series of morphological closings. The differences in size were normalized in the same way. We did not apply 9×9 squares to the closing because the SIPPER images are so small that most of them are diminished after the closing with 7×7 square as the structure element.

3.4 Domain specific features

Moment invariants and granulometries are general features, which can only capture some global information. They are far from enough to classify SIPPER images. Given advice from domain experts, we developed some domain specific features to help classification. The domain specific features include size, convex ratio, transparency ratio, ratio between the two eigenvalues, and ratio between the plankton's head and tail.

- Size: Size is the area of the plankton body, that is, the number of foreground pixels in the plankton image.
- Convex ratio: We implemented a fast algorithm [1] to get the convex hull of the plankton image. The convex ratio is the ratio between the plankton image size and the area of the convex hull. This feature contains information about the plankton boundary irregularity.
- Transparency ratio: This is the ratio between the area of the plankton image and the area of the plankton after filling all inside holes. The transparency ratio helps in recognizing the transparent plankton.
- Ratio between the two eigenvalues: Since some plankton are linear we first calculated the two eigenvalues of the body. Then the ratio between them was computed.
- Ratio between the head and the tail: Some plankton such as larvaceans have a large head relative to their tail. We computed the ratio between the head and tail to differentiate them. To do this we first rotated the image to make the axis with the bigger eigenvalue parallel to the x-axis. Assuming the smallest and largest x values are 0 and T respectively, we accumulated the number of foreground pixels along the x-axis from 0 to $\frac{1}{4}T$ and from $\frac{3}{4}T$ to T respectively. Then we took the ratio between them as the ratio between the head and the tail.

4 Support vector machines and probability model

Support vector machines (SVMs) [16] are receiving increasing attention these days and have achieved very good accuracy in pattern recognition, text classification, etc. [4]. In this section we describe SVMs and introduce a way to assign a probability value after multi-class SVM classification.

4.1 Support vector machines

In binary classification, SVMs try to find a hyperplane to separate the data into two classes. In the case in which all the data are well separated, the margin is defined as two times the distance between the hyperplane and the closest example. SVMs search for the hyperplane with the largest margin, which provides good generalization ability based on Vapnik' s VC dimension theory [16]. To increase the classification ability, SVMs first map the data into a higher dimension feature space with $\phi(x)$, then use a hyperplane in that feature space to separate the data. In the feature mapping stage, the kernel $k(x,y) =< \phi(x) \cdot \phi(y) >$ is used to avoid explicit inner product calculation in the high-dimension feature space. C-SVM, a typical example of soft SVMs, is described as follows. The slack variable ξ_i is used to handle non-separable examples.

Training set: there are m examples: $x_1, x_2, ..., x_m$ with class label $y_i \in$ {-1,1}.

C-SVM:

$$\text{minimize } \frac{1}{2} < w, w > + \frac{C}{m} \sum_{i=1}^{m} \xi_i \tag{3}$$

$$\text{subject to: } y_i(< w, \phi(x_i) > + b) \geq 1 - \xi_i \tag{4}$$

where w is normal to the hyperplane, C is a scalar value that controls the trade off between the empirical risk and the margin length, ξ_i is the slack variable and $C, \xi_i > 0$.

The decision function is $f(x) = \sum_i \alpha_i k(x_i, x) + b$, where α_i and b are computed from Eq. (3) and (4).

The Karush-Kuhn-Tucker condition of the optimal solution to Eq. (3) and (4) is:

$$\alpha_i(< w, \phi(x_i) > + b - 1 + \xi_i) = 0 \tag{5}$$

The α_i is nonzero only when Eq. (6) is satisfied. In this case the x_i contributes to the decision function and is called a support vector (SV).

$$y_i(< w, \phi(x_i) > + b) = 1 - \xi_i \tag{6}$$

Therefore, we get a sparse solution of the decision function, where only SVs contribute.

There are two main approaches to extending SVMs to multi-class classification. One-vs-all or One-vs-one used here (All possible groups of 2 classes are used to build binary SVMs.). It is tractable here, requiring 15 classifiers in the 6 class case.

4.2 Assigning probability values in support vector machines

A probability associated with a classifier is often very useful and it gives some confidence about the classification result. For instance, the classifier could reject the example and leave it to a human to classify it when the confidence is very low. Platt [11] introduced the sigmoid function as the probability model to fit $P(y = 1|f)$ directly. The parametric model is shown in Eq. (7).

$$P(y = 1|f) = \frac{1}{1 + exp(Af + B)} \tag{7}$$

where A and B are scalar values. f is the decision function of the binary SVM.

The A and B are fit with maximum likelihood estimation from the training set. Platt tested the model with 3 data sets including the UCI Adult and two other web classification data sets. The sigmoid-model SVM had good classification accuracy and probability quality in his experiments.

We followed the sigmoid model and extended it to the multi-class case. In the one-vs-one multi-class SVM model, since it is time consuming to do the parameter fitting for all $\frac{N(N-1)}{2}$ binary SVMs, we developed a practical method to compute the probability value while avoiding parameter fitting.

1. We assume $P(y = 1|f = 0) = P(y = -1|f = 0) = 0.5$. It means that a point right on the decision boundary will have 0.5 probability of belonging to each class. We get rid of parameter B in this way.

2. Since each binary SVM has a different margin, a crucial criterion in assigning the probability, it is not fair to assign a probability without considering the margin. Therefore, the decision function $f(x)$ is normalized by its margin in each binary SVM. The probability model of SVMs is shown as following.

$$P_{ij}(y = 1|f) = \frac{1}{1 + exp(\frac{-Af}{\|w\|})} \quad (8)$$

$$P_{ij}(y = -1|f) = 1 - P_{ij}(y = 1|f) = P_{ji}(y = 1|f) \quad (9)$$

   P_{ij}: binary SVM on class i vs. class j, class i is +1 and class j is -1

3. After we get the probability value for each binary SVM, the final probability for class i is computed as follows:

$$P(i) = \prod_{i}^{i \neq j} P_{ij}(y = 1|f) \quad (10)$$

   Normalize $P(i)$ to make $\sum_i P(i) = 1$

4. output $k = arg\ max_i P(i)$ as the prediction.

A is determined through numeric search based on the cost function $\sum_i logP(k)$ from 10-fold cross validation. After we finish learning a SVM model and set up a rejection threshold t, we reject an example and leave it to be classified by a person if $max_i P(k) < t$.

5 Experiments

Several experiments have been done to test our system. The Libsvm [2] support vector machine software was modified and used in our experiments. Libsvm uses decomposition in its optimization and a one-vs-one approach to do multi-class classification. We modified libsvm to produce a probabilistic output. In all experiments the gaussian radial basis function (RBF) was used as the kernel.

The gaussian RBF kernel: $k(x, y) = exp(-\frac{\|x-y\|^2}{2\sigma^2})$ where σ is a scalar value.

5.1 Initial experiments

The first training set has total 1285 SIPPER images ($50\mu m$ resolution). There are 64 diatoms, 100 protoctista, 321 doliolids, 366 larvaceans, and 434 Trichodesmium. We used C-SVM module with parameters $C = 200$ and $\sigma = 0.03$. To evaluate the accuracy of SVMs, we also compared it with a cascade correlation neural network [5] and a C4.5 decision tree with the default pruning settings [12]. Figure 2 shows the average accuracy of the three learning algorithms from 10-fold cross validation. A paired-t test was used to compare the results at the 95% confidence interval. The SVM is more accurate than the other two learning algorithms at the 95% confidence level.

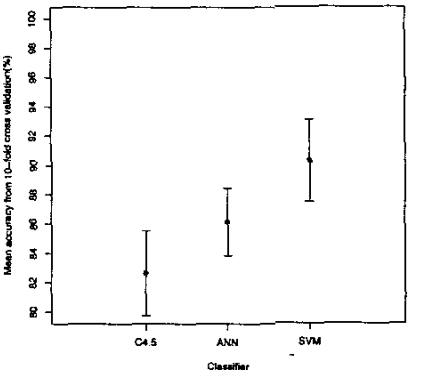

Figure 2: The mean accuracy and the range of standard deviation of the C4.5, the cascade correlation neural network (ANN) and the SVM from 10-fold cross validation on the 1285 SIPPER image set.

Table 1 shows the confusion matrix of the SVM from a 10-fold cross validation experiment. The overall average accuracy is 90.04%. While we had greater than 84% accuracy on most plankton, we only achieved 79% accuracy on the diatom class. The reason is that we only have 64 diatom samples in our training set and the SVM favors the class with more samples. For instance, given there is an overlap in the feature space between two classes: one with many examples and one with few examples. It is likely that most examples within that overlap come from the class with more examples. To minimize (3), the decision boundary is pushed away from the class with more examples and thus favors that class.

5.2 Experiments with unidentifiable particles

Encouraged by the initial experiment, we chose plankton images from some other collections of sample images. We picked the five most abundant types of plank-

Table 1: Confusion matrix from a 10-fold cross validation on 1285 SIPPER images with all 29 features. P, Di, Do, L and T represent Protoctista, Diatom, Doliolid, Larvacean and Trichodesmium respectively.

Classified	as P	as Di	as Do	as L	as T
P	84.4%	1.6%	9.4%	4.7%	0.0%
Di	2.0%	79.0%	11.0%	6.0%	2.0%
Do	0.8%	0.3%	92.8%	3.1%	0.0%
L	0.8%	0.3%	4.4%	88.0%	6.6%
T	0.0%	0.5%	0.2%	6.2%	93.1%

ton, which account for 95% of the plankton samples from the particular area of acquisition in the Gulf of Mexico. They are copepods, doliolids, larvaceans, protoctista and Trichodesmium. The image quality in this training set is not as good as in the initial experiment. Some information, unknown to us, was used by ocean experts to label the images. Also, we were forced to handle unidentifiable particles in this experiment.

There were a total of 6000 images: 1000 images of each plankton class and 1000 unidentifiable particles. We used C-SVM with $C = 200$ and $\sigma = 0.032$. Figure 3 shows the average accuracy of three learning algorithms from 10-fold cross validation. A paired-t test was used to compare the results at the 95% confidence interval. The SVM is more accurate than the other two learning algorithms at the 95% confidence level.

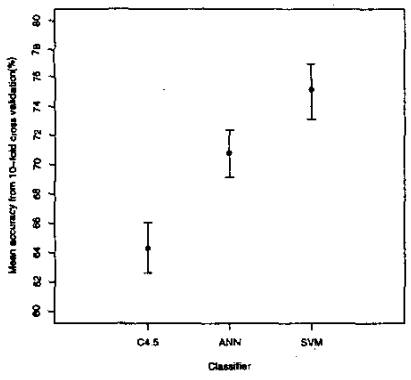

Figure 3: The mean accuracy and the range of the standard deviation of the C4.5, the cascade correlation neural network (ANN) and the SVM from 10-fold cross validation on the 6000 SIPPER image set.

Table 2 shows the confusion matrix from a 10-fold cross validation. The overall average accuracy is 75.12%. The average accuracy from the five types of plankton is 78.56%.

There were a significant number of larvaceans confused with Trichodesmium. This observation disagrees with the first experiment where we had high classification accuracy for both types of plankton. The justification is that some larvacean and Trichodesmium are linear objects. Domain experts know that there are some ocean areas where larvacean or Trichodesmium are less common. They labeled the linear objects as larvacean or Trichodesmium because they knew the other plankton were less commonly found in the particular ocean areas examined. Therefore, there are many linear particles without significant features to differentiate between the two types of plankton in this training set, thus dropping the classification accuracy on larvaceans and Trichodesmium.

Table 2: Confusion matrix from a 10-fold cross validation on 6000 SIPPER images with all 29 features. C, D, L, P, T, and U represent Copoped, Doliolid, Larvacean, Protoctista, Trichodesmium and Unidentifiable particles respectively.

Classified	as C	as D	as L	as P	as T	as U
C	84.2%	0.6%	3.1%	1.0%	5.5%	5.6%
D	0.2%	82.9%	2.4%	8.7%	0.4%	5.4%
L	3.2%	1.9%	68.8%	1.4%	11.1%	13.6%
P	1.7%	5.3%	1.1%	84.4%	3.1%	4.4%
T	3.3%	0.6%	9.4%	1.8%	72.5%	12.4%
U	4.3%	3.1%	15.8%	5.4%	13.5%	57.9%

5.3 Probability assignment experiments

We used the same training set as in the last experiment with a 15-feature subset, with equivalent discrimination ability whose creation details are omitted due two space limitations. To determine A in Eq. (7), we varied the A value in 10-fold cross validation experiments and picked the one with the lowest cost $\sum_i logP(k)$. The best value for A in our experiment is 1500. We drew a rejection curve by varying the rejection threshold. Figure 4 shows that the accuracy goes up as the rejection ratio increases, which is reasonable.

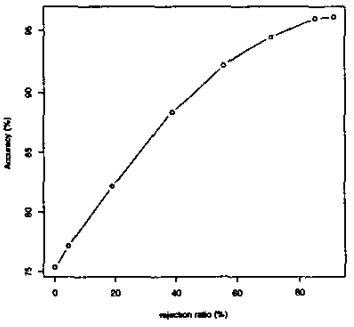

Figure 4: Rejection curve-Overall accuracy vs. rejection rate.

6 Conclusions and future work

This paper presents a plankton recognition system for binary SIPPER images. General features as well as domain specific features were extracted and a support vector machine was used to classify examples. We also developed a new way to assign a probability value after the multi-class SVM classification. We tested our system on two different data sets. The recognition rate exceeded 90% in one experiment and was greater than 75% on the more challenging data set with unidentifiable particles. SVM is more accurate than the C4.5 decision tree and the cascade correlation neural network at the 95% confidence level on the two data sets.

The system did not do well at recognizing unidentifiable particles. It is hard to develop specific features to describe the unidentifiable particles because they vary so much. More powerful descriptive and robust general features seem needed in our future work. Recently, an advanced SIPPER system has been developed to produce grayscale SIPPER images at 25 μm resolution. We are in the process of developing methods and features for higher resolution (25 um resolution) grayscale SIPPER images[1].

References

[1] Mark De Berg(Editor), Marc Van Kreveld, Mark Overmars, O. Schwarzkopf, Mark de Berg, *Computational geometry: algorithms and applications*, Springer, 2001.

[2] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: a library for support vector machines", http://www.csie.ntu.edu.tw/ cjlin/papers/libsvm.pdf, 2002.

[3] A. Ciobanu and H. D. Buf, "Identification by contour profiling and legendre polynomials", *Automatic diatom identification*, pp. 167–186, World Scientific, 2002.

[4] N. Cristianini, J. Shawe-Taylor, *Introduction to support vector machines and other kernel-based learning methods*, Cambridge University Press, 2000.

[5] S. E. Fahlman and C. Lebiere, "The cascade-correlation learning architecture", Advances in neural information processing systems, Vol 2, pp. 524–532, 1991.

[6] S. Fischer and H. Bunke, "Identification using classical and new features in combination with decision tree ensembles", *Automatic diatom identification*, pp. 109–140, World Scientific, 2002.

[1] Acknowledgments: The ratio between two eigenvalue and the ratio between head and tail features were suggested by Xiaoou Tang (personal communication). The research is partially supported by the U. S. Navy, Office of Naval Research, under grant number N00014-02-1-0266 and the NSF under grant EIA-0130768.

[7] M. K. Hu, "Visual pattern recognition by moment invariants", IRE Trans. Information theory, Vol IT, No. 8, pp. 179–187, 1962.

[8] R. E. Loke and H. d. Buf, "Identification by curvature of convex and concave segment," *Automatic diatom identification*, pp. 141–166, World Scientific, 2002.

[9] G. Matheron, *Random sets and integral geometry*, John Wiley and Sons: New York, 1975.

[10] I. Pitas, *digital image processing algorithms and applications*, John Wiley and Sons, Inc., 2000.

[11] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods", Advances in Large Margin Classiers, pp. 61–74, 2000.

[12] J. R. Quinlan, *C4.5: Programs from empirical learning*, Morgan Kaufmann, San Francisco, 1993.

[13] S. Samson, T. Hopkins, A. Remsen, L. Langebrake, T. Sutton, and J. Patten, "A system for high resolution zooplankton imaging", IEEE journal of ocean engineering, Vol 26, No. 4, pp. 671–676, 2001.

[14] L. M. Santos and H. D. Buf, "Identification by gabor features," *Automatic diatom identification*, pp. 187–220, World Scientific, 2002.

[15] X. Tang, W. K. Stewart, L. Vincent, H. Huang, M. Marra, S. M. Gallager and C. S. Davis, "Automatic plankton image recognition," Artificial intelligence review, Vol 12, No. 1-3, pp. 177–199, 1998.

[16] Vladimir N. Vapnik, *The nature of statistical learning theory*, Springer, 2000.

[17] M. H. F. Wilkinson, A. C. Jalba, E. R. Urbach, and J. B. T. M. Roerdink, "Identification by mathematical morphology," *Automatic diatom identification*, pp. 221–244, World Scientific, 2002.

[18] C. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curve", IEEE transaction on computers, Vol C, No. 21, pp. 269–281, 1972.