MVA2007 IAPR Conference on Machine Vision Applications, May 16-18, 2007, Tokyo, JAPAN

One Fish, Two Fish, Butterfish, Trumpeter: Recognizing Fish in Underwater Video

Andrew Rova*
Simon Fraser University
Burnaby, BC
arova@cs.sfu.ca

Greg Mori*
Simon Fraser University
Burnaby, BC
mori@cs.sfu.ca

Lawrence M. Dill†
Simon Fraser University
Burnaby, BC
ldill@sfu.ca

Abstract

This paper presents a deformable template object recognition method for classifying fish species in underwater video. This method can be a component of a system that automatically identifies fish by species, improving upon previous works which only detect and track fish and those that rely on significant inter-species shape variations or special equipment. Our method works with video shot by a standard uncalibrated camera in a natural setting rather than the calibrated stereo cameras and man-made imaging environments described in other publications. We use deformable template matching which employs an efficient combination of shape contexts and large-scale spatial structure preservation. Experimental results demonstrate the improvement of deformable template matching over raw SVM texture-based classification.

1 Introduction

Quantifying the number of fish in a local body of water is of interest for applications such as guiding fisheries management, evaluating the ecological impact of dams, and managing commercial fish farms. Going beyond an aggregate count of aquatic animals, information about the distribution of specific species of fish can assist biologists studying issues such as food availability and predator-prey relationships [11]. Applications like these motivate the development of methods for collecting biological data underwater.

Besides video, other options for automating underwater fish counting include devices employing hydroacoustics (sonar), resistivity counters and infrared beams [5]. Of the alternatives, sonar is best suited to coarser detections such as finding shoals of fish, while resistivity and infrared counters require fish to swim through relatively narrow enclosed spaces. Underwater video is a non-intrusive method of counting fish, as well as the only one of these techniques that can classify fish by species based on textural appearance. Other attempts to visually identify fish by species rely on constrained images or shape to make distinctions [5, 14, 16]. None of these methods would work for the problem described in this paper, where the two species of interest have very similar shapes and the environment is natural.

For computer vision researchers, this problem presents a number of interesting challenges. First, the complex environment confounds simpler approaches like luminance thresholding and background subtraction. Issues include shifting colors, uneven and variable illumination, sediment in the water and undulating underwater plants. Secondly, the recognition task is non-trivial; common cues such as background context, distinctive colors and unique shapes are absent. Fig. 1 shows examples of the two species of fish we attempt to discriminate between. Finally, the fish appear in a variety of scales, orientations, and body poses, all factors that complicate recognition.

We approach this task as a deformable template matching problem followed by the application of a supervised learning classifier. Aligning the images before classifying by appearance provides a demonstrable increase in performance. A primary contribution of this paper is the novel combination of shape context descriptors [3] with efficient dynamic programming-based correspondence using the distance transform [7] for deformable template matching. This allows the estimation of template-to-query correspondences which would not be possible using shape contexts alone because of the low quality of the underwater video images. Tree-structured dynamic programming and fast distance transform techniques from [7] make it computationally feasible to simultaneously consider both shape context matching costs and points' spatial relationships, and to find a globally optimum correspondence. Our method recovers correspondences in low-quality images that lack distinctive or stable features; it is similarly motivated but computationally cheaper than the IQP approach in [4].

1.1 Previous work

The idea of deformable template matching has deep roots within the computer vision community. Fischler and Elschlager [9] developed a technique based on energy minimization in a mass-spring model. Grenander et al. [10] developed these ideas in a probabilistic setting. Yuille [22] developed another variant of the deformable template concept by means of fitting hand-crafted parametrized models.

Other approaches in this vein [13] first attempt to find correspondences between a pair of images prior to an

*School of Computing Science, Vision and Media Lab
†Department of Biological Sciences, Behavioural Ecology Research Group

appearance-based comparison, as we do in this paper.

Recent years have seen the emergence of part-based models approaches [15, 8, 6, 2] that characterize appearance using a collection of local image patches selected by interest point operators. Shape information is encoded via spatial relationships between the local patches. However, for our problem interest point operators would not be successful due to the lack of distinctive and stable image features. Thayananthan et al. [19] added figural continuity constraints to shape context matching of contours. In this work, we use tree-structured spatial constraints that can be efficiently optimized using the distance transform.

Images of a constrained environment, for example from a glass-walled fish ladder, are used to in other underwater fish data-analyses such as shape-based classification [14] or counting after background-subtraction [16]. Other methods use stereo cameras to estimate traits such as fish mass [20], employ special devices through which fish must swim to generate a silhouette for shape classification [5], or utilize color for fish recognition [18]. Undersea animals are detected and tracked in a natural setting in [21], however identification is not performed.

2 Approach

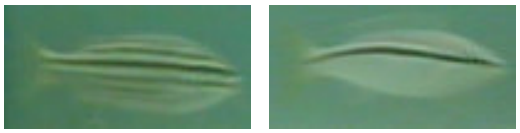
Our goal is to distinguish between the fish species shown in Fig. 1. The Striped Trumpeter (Fig. 1(a)), has multiple horizontal markings while the Western Butterfish (Fig. 1(b)) sports a single bold stripe. Since the images' color information is dominated by the water's hue, color is not useful to differentiate these fish types. Shape also provides little discernement, so we will focus on texture-based classification.

We use deformable template matching to align template images and query images in an attempt to improve the performance of such a texture-based classifier, whose results are sensitive to pixel alignment. The following sections describe the details of this approach.

2.1 Model generation

The following steps are repeated for each of the two classes. First, a template image representative of the current fish class is chosen, e.g. Fig. 1(a) or Fig. 1(b). A set of edge points similar to Fig. 2(f) are extracted from the template using Canny edge detection. Next, a subset of 100 template edges is randomly chosen from the set of edge points. The size of this subset was chosen empirically based on our previously fixed image size. The edge subset is then connected into a minimum spanning tree (MST) using Prim's algorithm. An example of a template overlaid with a MST is shown in Fig. 2(a). These model trees are stored and reused for the remainder of the matching process.

Finding the best match of a template tree model in a query image will be phrased as an optimization problem with two cost terms to be minimized. As in [7], if a tree has n vertices $\{v_1, \ldots, v_n\}$ and an edge $(v_i, v_j) \in E$ for each pair of connected vertices, then a configuration



(a) Striped Trumpeter (b) Western Butterfish

Figure 1: The two types of fish to be classified.

$L = (l_1, \ldots, l_n)$ gives an instance of an object, where each l_i specifies the location of part v_i. Then, the best match of a model to a query is

$$L^* = \arg\min_L \left(\sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i,v_j)\in E} d_{ij}(l_i, l_j) \right) \quad (1)$$

where $m_i(l_i)$ is a matching cost between features in the model and query image at location l_i, and $d_{ij}(l_i, l_j)$ is the amount that model edge (v_i, v_j) is changed when vertex v_i is located at l_i and v_j is placed at l_j. In our method, $m_i(l_i)$ will be the shape context matching cost defined in §2.2.1, and $d_{ij}(l_i, l_j) = \|(l_j - l_i) - (l_j^m - l_i^m)\|_2$, with l_k^m denoting the location of vertex v_k in the model tree.

This means that the best set of point correspondence mappings from the template into the query are those that minimize the total shape context matching cost and at the same time least alter the relative spatial locations of vertices which are neighbors in the model tree.

2.2 Deformable template matching

For the deformable template matching we combine the strengths of shape context descriptors [3] with the distance transform methods of [7]. Rather than matching a set of edge points in the model image with another set of edge points in the query image, we search every pixel location in the query image and thus find a global optimum match for the model tree.

2.2.1 Shape contexts

Our method employs shape contexts as image features because they are well suited to capturing large-scale spatial information in images exhibiting sparse edges, a common characteristic of our underwater images.

Shape contexts [3] (SCs) are coarse radial histograms of image edge points. For a particular location, its shape context captures the relative spatial locations of all the edge points within the circumference of the shape context bins. In this work we use generalized shape contexts [17] which capture the dominant orientation of edges in each bin, rather than just the point counts. For a point p_i, the shape context is a histogram h_i capturing the relative distribution of all other points such that

$$h_i(k) = \sum_{q_j \in Q} t_j, \text{ where } Q = \{q_j \neq p_i, (q_j - p_i) \in \text{bin}(k)\}, \quad (2)$$

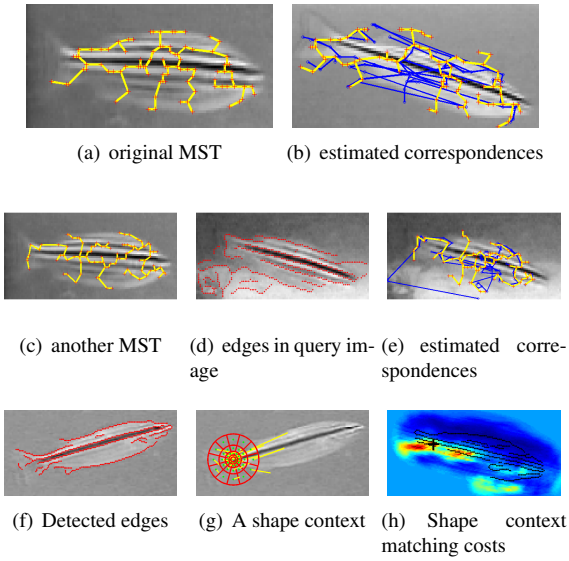
and t_j is a tangent vector that is the direction of the edge at q_j. Figs. 2(f) and 2(g) show a visualization of edge

points and a SC. When comparing two shape contexts, we treat them as feature vectors and compute the L_2 distance between them. This distance is referred to as the shape context matching cost; in our method, this is the first minimization term $m_i(l_i)$ in Eq. (1).

After a MST is constructed in the template image, shape contexts are calculated at each of the points which make up the vertices of the model tree, and these will be matched with shape contexts computed at every pixel location in the query image.

2.2.2 Distance transforms

Distance transforms and dynamic programming on a tree structure [7] make it computationally feasible to find globally optimal correspondences between the template model and an unknown query image, a situation for which the methods of [3, 4] are ineffective or intractable. In particular, the distance transform method can find the global optimum of Eq. (1) in time $O(nt)$, where n is the number of pixels and t is the number of nodes in the tree. This allows efficient computation of $d_{ij}(l_i, l_j)$ in Eq. (1).



(a) original MST (b) estimated correspondences

(c) another MST (d) edges in query image (e) estimated correspondences

(f) Detected edges (g) A shape context (h) Shape context matching costs

Figure 2: Correspondences estimated using SC costs and spatial structure (thicker yellow lines with crosses) are better than spatially incoherent estimates based on SC costs alone (crisscrossing narrower blue lines), as shown in 2(b). 2(e) shows how spatial structure helps deal with clutter. Canny edges are shown in 2(f). 2(g) visualizes a shape context histogram; yellow bars represent the magnitude and direction of the histogram bins. 2(h) visualizes shape matching costs at every pixel of a new image, with a cross at the best match point.

.

2.2.3 Iterative warping

The techniques of [7] are then employed to find L^* from Eq. (1)—the globally optimum configuration of a template model tree in the query image. From the template-to-query correspondence estimates, a least-squares affine transformation from the query to the template can be derived. The use of affine transformations is justified since the fish are relatively flat, and since in practice the video sequence usually contains at least one side-on image of each fish. This transformation is then applied to the edge points from the query image, shape contexts are recomputed everywhere in the query image, and the correspondence process is repeated. For our experiments, a maximum of 4 iterations were performed; if a reasonable affine transformation is not found, the iterative warping process aborts. After the iterative transformation of the query edge points, the complete estimated transformation is applied to the query image. Fig. 3 shows some examples of warped images.

2.3 Texture-based classification

Once the query images have been transformed into estimated alignment with the template they are processed to extract texture properties. First, each image is convolved with a 3-pixel-tall vertical central difference kernel. The motivation for vertical derivative filtering is that after successful warping, the vertical direction captures the most image information. Next, the filter response is half-wave rectified to avoid cancellation during subsequent spatial aggregation. Each half-wave component of the filter response is summed into 7-pixel square sections. Finally, all of the combined filter responses are concatenated into a feature vector as input for the classifier.

SVMs are binary classifiers. However, in our method there are two templates, one for each type of fish, and each query image is warped to both templates. This means that we have two SVMs whose outputs need to be combined to get a final classification decision. Our situation is a simplified version of the multi-SVM problem of [1]. If both SVMs agree on a classification decision, then all is well. If the two SVMs assert opposite classifications, then the decision of the SVM with the greater absolute distance to its separating hyperplane is taken to be the true one.

3 Results

The steps described in §2—tree-structured model generation, point-correspondence estimation, iterative warping and SVM texture classification—were implemented in MATLAB and tested on a set of manually cropped underwater video images of two fish species. We used *svmLight* [12] for SVM classification. For both species of fish being classified 160 images were manually cropped from frames of underwater video. In these 320 images, the fish appear at different angular poses although all of their heads face the right. All images were converted to grayscale, de-interlaced and resized to 50 $\times$ 100 pixels, empirically chosen based on the size of the majority of fish in the images.

8-fold cross validation on a training set consisting of half the image data was used to select the best SVM ker-

Table 1: Results of SVM classification

SVM kernel	unwarped	warped
`linear`	84%	90%
`polynomial`	81%	86%

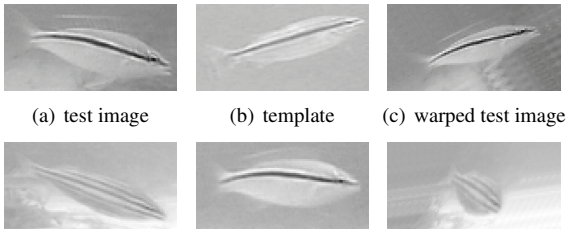
nels and parameters. SVMs with these attributes were then constructed for the entire training set. The results of running these SVMs on the set of test images are reported in Table 1. The basis of comparison is the accuracies of SVMs trained on texture features from the original, unwarped images.

For both the `linear` and `polynomial` SVM kernels, warping the images into alignment with a template prior to classification improved the classification accuracy, in the best case by up to 6% (90% versus 84%).

4 Conclusion

This work describes a novel combination of existing techniques applied to classifying fish by textural appearance in underwater video. Other methods, such as the linear assignment problem (used in [3]) or IQP (used in [4]),would be ineffective or computationally intractable in this setting. In addition, our work goes beyond previous fish tracking, counting and classification methods by identifying similarly shaped fish species based on textural appearance.

As future work, we are developing a preprocessing and tracking component that will output cropped fish images to be classified using the method described in this paper. The goal is a complete system that automatically detects, tracks, counts and classifies fish in underwater video, without requiring manual cropping of fish images.



(a) test image (b) template (c) warped test image

(d) test image (e) template (f) warped test image

Figure 3: Warping examples: 3(c) and 3(f) contrast successful and unsuccessful recoveries of template-image correspondences.

Acknowledgments

Funding provided by CFI and BCKDF as part of the SDATS project. This is Contribution #24 of the Shark Bay Ecosystem Research Project (SBERP).

References

[1] E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. In *Proc. ICML*, pages 9–16, 2000.

[2] Y. Amit, D. Geman, and K. Wilder. Joint induction of shape features and tree classifiers. *IEEE Trans. PAMI*, 19(11):1300–1305, November 1997.

[3] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. PAMI*, 24(4):509–522, 2002.

[4] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *Proc. CVPR '05*, pages 26–33, 2005.

[5] S. Cadieux, F. Lalonde, and F. Michaud. Intelligent system for automated fish sorting and counting. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2000.

[6] G. Dorko and C. Schmid. Selection of scale invariant neighborhoods for object class recognition. In *Proc. 9th Int. Conf. Computer Vision*, pages 634–640, 2003.

[7] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, 2005.

[8] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. CVPR '03*, volume 2, pages 264–271, 2003.

[9] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. Computers*, C-22(1):67–92, 1973.

[10] U. Grenander, Y. Chow, and D. Keenan. *HANDS: A Pattern Theoretic Study Of Biological Shapes*. Springer, 1991.

[11] M. R. Heithaus and L. M. Dill. Food availability and tiger shark predation risk influence bottlenose dolphin habitat use. *Ecology*, 83(2):480–491, 2002.

[12] T. Joachims. *Advances in Kernel Methods–Support Vector Learning*, chapter 11: Making Large-Scale SVM Learning Practical. MIT-Press, 1999.

[13] M. Lades, C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. Computers*, 42(3):300–311, March 1993.

[14] D. J. Lee, S. Redd, R. Schoenberger, X. Xu, and P. Zhan. An automated fish species classification and migration monitoring system. In *Proc. 29th Annual Conf. IEEE Ind. Elect. Soc.*, pages 1080–1085, November 2003.

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[16] E. F. Morais, M. Campos, F. L. Pádua, and R. Carceroni. Particle filter-based predictive tracking for robust fish counting. In *Proc. SIBGRAPI*, 2005.

[17] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *IEEE Trans. PAMI*, 27(11):1832–1837, November 2005.

[18] N. J. C. Strachan. Recognition of fish species by colour and shape. *IVC*, 11(1):2–10, 1993.

[19] A. Thayananthan, B. Stenger, P. H. S. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *Proc. CVPR '03*, pages 127–133, 2003.

[20] R. Tillett, N. McFarlane, and J. Lines. Estimating dimensions of free-swimming fish using 3D point distribution models. *CVIU*, 79:123–141, 2000.

[21] D. Walther, D. R. Edgington, and C. Koch. Detection and tracking of objects in underwater video. In *Proc. CVPR '04*, 2004.

[22] A. Yuille. Deformable templates for face recognition. *J. Cognitive Neuroscience*, 3(1):59–71, 1991.