Vision Recognition Using Shape Context for Autonomous Underwater Sampling

Katherine McBryan, Dr. David L. Akin
Space Systems Lab
University of Maryland
College Park, Maryland

***Abstract*—**

The ocean floor is one of the few remaining unexplored places on the planet. Underwater vehicles, both teleoperated and autonomous, have been built to take images of the ocean floor. The depth that a teleoperated vehicle can achieve is limited by its tether. Autonomous vehicles are able to study the deepest parts of the ocean without a complex tether system. These vehicles, while being great at mapping the ocean floor, are not able to autonomously retrieve samples. In order to retrieve samples the vehicle must: know what objects look like, correctly identify new instances of the target object, estimate the pose so the manipulator can grab it, and retrieve its coordinates in 3D space. Color filtering, shape context and the use of stereovision have been used to autonomously locate, identify, and estimate the pose of objects. Color filtering allows the image to be filtered so that only objects of similar color remain and extraneous information can be disregarded.

Shape context matches the shape, as defined by the edge pixels, of each potential target to a known object. Shape context uses a costing function to determine if the potential target is a match to the known object. The costing function takes into account the amount of 'bending energy' it takes to make the shape of the potential target conform to that of the known object. This gives a metric of how well the match is between the potential target and a known object and is done for both the left and right cameras. Once objects have been identified in each image, calibration parameters can be used to retrieve the 3D position of the object. This allows a manipulator on an underwater vehicle to autonomously sample targets.

***Index Terms*—Autonomous Undersea Vehicle, Shape Context, Object Recognition, Sampling.**

I. Introduction

Autonomous underwater vehicles (AUVs) are a great asset to the research community. They have proven themselves to be invaluable in extreme environments and in imaging, though often they leave researchers wanting more. The ability to take samples is extremely important – and very difficult. In order to autonomously take samples the vehicle must have a system for recognizing objects, locating the samples, and grabbing the samples. The Subsea Arctic Manipulator for Underwater Retrieval and Autonomous Interventions (SAMURAI) manipulator was developed by the Space Systems lab at the University of Maryland as part of the NASA Astrobiology Science and Technology for Exploring the Planets (ASTEP) program and further supported by the National Science Foundation (NSF) after cancellation of the ASTEP program. SAMURAI is an autonomous manipulator designed to withstand pressures at a depth of 6,000m. The manipulator has an end effector which allows it to pick up samples and place them in sample containers. SAMURAI is designed to be mounted to JAGUAR, an AUV, developed by the Woods Hole Oceanographic Institute (WHOI). [4,7]

The Autonomous Vision Application for Target Acquisition and Ranging (AVATAR) is the vision system for SAMURAI. AVATAR has a stereovision system consisting of two Point Grey Scorpions cameras. AVATAR is used to recognize an object of interest in an image, an object which has been previously identified by scientists, determine the distance to the object of interest, and determine the pose so that the manipulator is able to grasp it. In order to recognize an object of interest, the object must first be observed and identified as important by scientist. This is accomplished by doing a preliminary survey of the area before a sample mission is underway. The preliminary survey will take images and/or video and allow scientist to identify which objects they would like samples taken of. AVATAR and SAMURAI are then used to identify and grab the specified object of interest. [3,7]

AVATAR uses the color, aspect ratio, density ratio and the total area to identify an object. A color filter is applied to each image in order to isolate an object of interest from the background. This is done by filtering the image for the interested object's color. This is done in hue, saturation and value (HSV) space rather than the more typical red, green, blue (RGB) space. The HSV color space has separate value for hue and saturation and is thus less affected by changes in illumination.

The aspect ratio is determined from a box defined by the top most left point and the bottom most right point of an object's edge. The area of the target is found by summing the pixels that make up the cluster enclosed by the box. The density ratio is found by dividing the object's area by the area of the box it is contained in. This results in the four terms that describe an object; a color value and three terms to determine the shape. A range of values for each of these terms is used to identify an object. Test images, with known instances of the

object, are used to find these ranges. Once an object is identified in both the left and right frame, stereovision is used to determine the distance.

This method of identification is not rotationally invariant. The aspect ratio of an object changes as it is rotated: for example, a square has a different aspect ratio than a diamond. In order to be able to account for some rotations, the identification ranges must be increased. This can result in false positives or fail to identify an object due to its rotation.

AVATAR's method uses the extreme points of the shape. Shape context, introduced by Belongie et al., is a method to identify objects based on the entire shape. One of the benefits of using shape context with thin plate splines (TPS) is that it is rotationally invariant. The use of shape context results in fewer false positives. [1,2,5]

II. Shape Context Overview

Shape context and TPS compares two objects; a reference and a target, by calculating how much 'energy' is required to morph the target to the reference object. A low 'energy' measurement is when the two objects are similar in shape – it would take little 'energy' to morph the target object to the reference. In order to morph one object to another, corresponding points between the reference and the test object must be found. The identification of an object is based on the cost to match points, the 'bending energy' require to morph the object to the reference, and a cost based on a simple comparison between the morphed target to the reference. [1,2,5]

Shape context uses radial histograms to match the object's edge points to those of the reference. At every sample edge point a histogram is created using a radial grid and bins to discretize the remaining edge points according to distance and angle. The image is translated such that the sample edge point is the origin of the radial histogram. Every other point is placed into a bin, see Fig. 1. The histograms between the same points in two different images of the same object are very similar. For example, the pixel at the nose of a Yellow Tang fish will have a very similar histogram to that of the pixel at the nose of a Yellow Tang fish in a second image. For a large object it is beneficial to reduce the number of edge points in order to reduce computation time.

In order to make the analysis rotational invariant, the object is rotated so that the tangent at each sample point is zero degrees. This angle was found by linearizing the points around the sample point. This only works in planar rotation, which is acceptable in this application as the robot is sampling from the sea floor. Rotations around other axis result in different points of view and very different edge properties.

A cost matrix is created to find which points match the best between a reference image and a sample image. The cost function takes into the account the differences in the histogram between two points as well as the difference in the tangent angle of each point. These terms are weighted so that the user is able to determine which is more important, see Eqn. 1. [2]

The Hungarian method is used to find which points should be matched in order to minimize the total cost between the two images. The Hungarian method does not allow the multiple points to be mapped to the same point. The total cost is the sum of cost between matching each point. A threshold is set to allow points to remain unmatched when the cost to match the point is too high. [2]

$$C(i,j)=(1-B)\left(\frac{1}{2}\sum_{k=1}^{k}\frac{\left(g(k)-h(k)\right)^2}{g(k)+h(k)}\right)+$$

$$(B)\left(\frac{1}{2}\right)\left\|\frac{\cos(\theta_1)-\cos(\theta_2)}{\sin(\theta_1)-\sin(\theta_2)}\right\| \qquad (1)$$

Where:

k = number of bins
h = reference histogram at point j
g = histogram at point i
θ1 = tangent angle at i
θ2 = tangent angle at j
B= weight

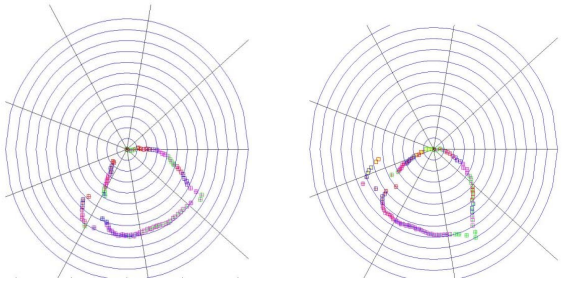

Figure 1: This is the outline of a Yellow Tang fish which has been extracted using a color filter. The images in this figure demonstrate how the edge is rotated and translated such that the sample point is at the origin with a tangent angle aligned with the x-axis.

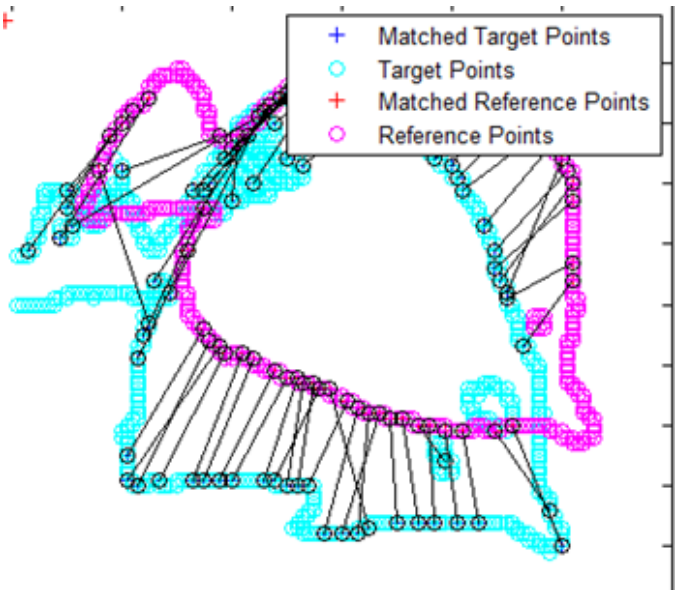

Figure 2: The edge points for a reference fish and a target fish are shown with the matched points between them connected. Note that many of the points remain unmatched. Each object has been scaled to same size. This allows the same radial grid to be used when creating histograms for each object.

Thin Plate Splines are then used to morph the target object to the reference using the points corresponding between the two objects found by minimizing the cost matrix. This method allows the deformation to be modeled and quantified. The method behind TPS can be thought of as taking the target and reference objects as thin pieces of metal and finding the amount of energy it would take to bend the target object to the reference object. The result is a measure of 'bending energy', of which a lower amount means it was easier to match the target to the reference, see Eqn. 2. TPS has the benefit of being able to be linearized, see Eqn. 3. [6]

$$I_f = \iint_{R^2} \left(\frac{\partial^2 f}{\partial x^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 f}{\partial y^2}\right)^2 dx\,dy \quad (2)$$

$$f(x,y) = a_1 + a_x x + a_y y + \sum_{i-1}^{n} w_i U\left(\left| (x_i, y_i) - (x,y) \right| \right)$$

$$\text{Where:} \quad U(r) = r^2 \log(r^2) \quad (3)$$

TPS maps the target object to the reference and results in a morphed image of the target object- as well as the amount of 'bending energy' necessary to achieve this. If the target image is very similar to the reference, it will take little energy to morph it to the reference and the resulting morph will be very similar to the reference. The third measure of how well the object of interest matches the reference is a simple comparison between the resulting morphed image and the reference. In this case a very simple comparison was done comparing the areas – though more sophisticated methods could also be used.

The shape context and TPS method results in three measures of how well the target matches the reference image. These are the cost to match points between the two images, the bending energy and the comparison between the morphed image and the reference. [1]

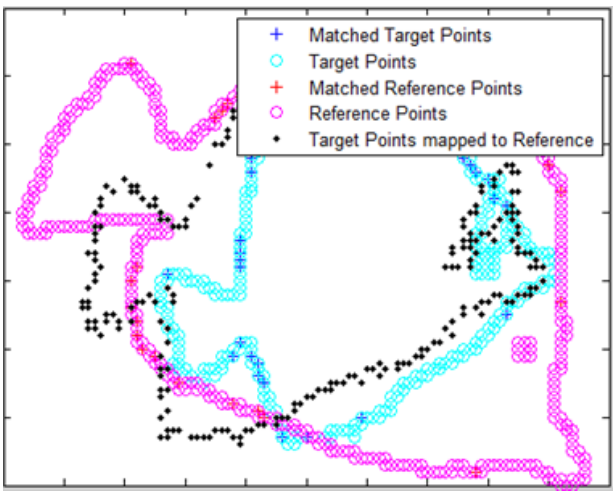

Figure 3: Shape context is a method for comparing points using histograms. Thin Plate Splines takes matching points between two objects and uses those points to morph an object to a reference. This image shows the resulting morph.

III. Analysis

The object used in testing was the Yellow Tang fish. A reference image is used to describe the object and then the AVATAR method and shape context are used to determine the location and number of Yellow Tang fish in multiple images. One of the test images did not include a Yellow Tang but a different species of yellow fish and another image included multiple species of fish. The Yellow Tang fish was selected due to the high of the contrast in color. The Yellow Tang fish also allows the natural variation in species to be taken into account. A real world sampling mission will include natural variations in the desired objects.

A Graphical User Interface (GUI) was developed in MATLAB to more efficiently apply the filters. This allowed the user to load two images, apply color filtering and use both methods of object recognition. The GUI allows the user to view two images, such as a reference and a test image, at the same time and allows the user to determine the appropriate ranges on all identifying variables.

A. AVATAR – Aspect Ratio Method

A color filter was applied to the reference image; this was done so that the desired Yellow Tang fish was isolated. The aspect ratio, area and density ratio of the reference were found. This was also done for a second reference image with a known Yellow Tang fish. This was used to create acceptable ranges for the color, aspect ratio, area, and density ratio.

Each test image had the color filter applied, as defined by the color ranges found from the two references. The results from the color filtering were then isolated into blobs, which were defined by touching pixels. Pixels were considered touching if they were in any of the surrounding 8 pixels (touching up, down, left, right and the diagonals). At this point the filtered image was reduced to grayscale for computation purposes. Any blob less than 3 pixels was discarded as noise. The edge points and size of each blob was found by counting the pixels and finding the upper left most point and the lower right most point. A rectangle was drawn using these two points. This allowed the aspect ratio, area and density ratio for each blob to be calculated. The blobs with an aspect ratio, area and density ratio that fit within the specified ranges was determined to be a Yellow Tang fish. [3]

B. Shape Context and TPS

The first step of applying the shape context and TPS method is to apply a color filter. This color filter is the same as that used by AVATAR. Any pixel that does not fit within the specified HSV ranges is discarded. The color ranges were adjusted until the Yellow Tang fish in the reference and test images were isolated and the edges were able to be found.

The pixels remaining after the color filter is applied are then grouped into blobs and, for the sake of computation time, reduced to grayscale values. The entire edge of each of these blobs is then found. This is done by checking if each pixel was

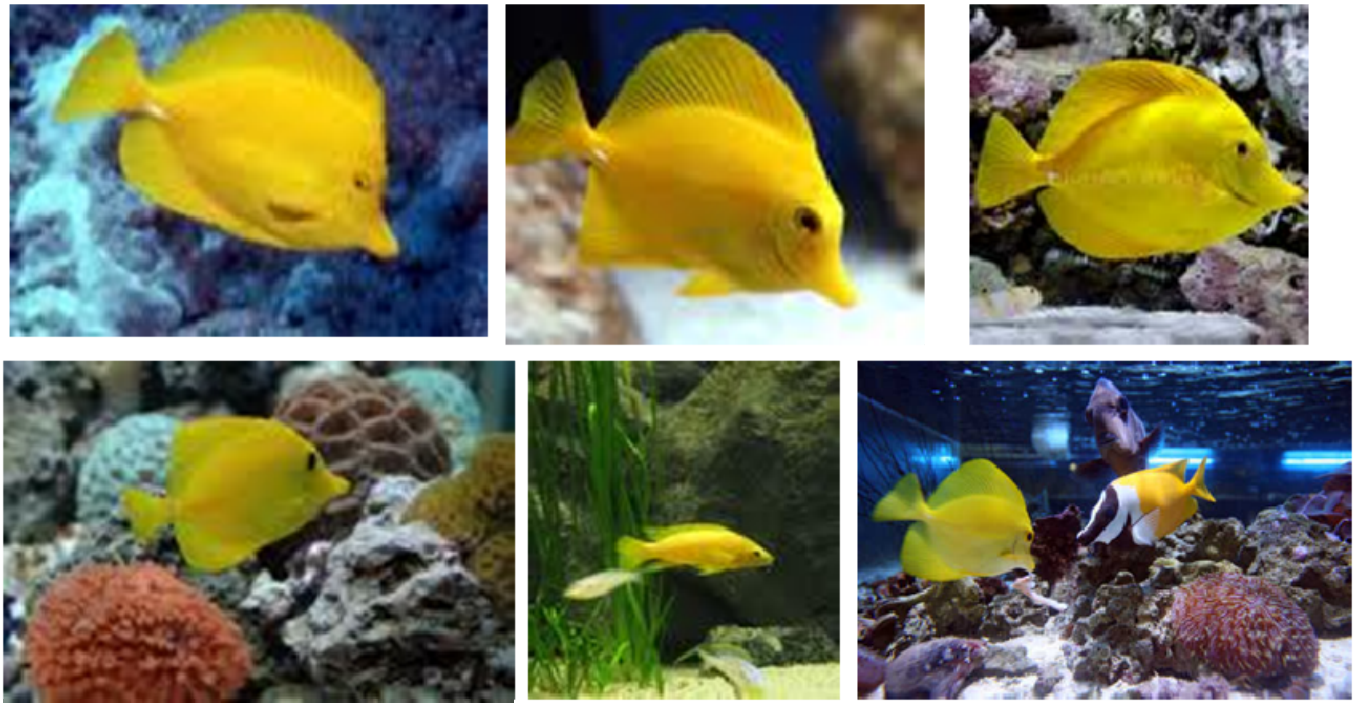

Figure 4: Images found using Google image to compare the performance of AVATAR to that of the shape context and TPS method. Reference image 1- top left (botany.hawaii.edu), Reference image 2-top center(yellowtangfish.blogspot.com) Images for comparison: Top right (liveaquaria.com), bottom left (canreef.com), lower center(celticangle.org), lower right (tampabay.com/news/business article1041507.ece)

surrounded or not. The number of edge pixels were then reduced to a user specified amount, in this case 100 pixels. This was done in order to reduce computation time. The sample edge points should be spaced such that the shape is maintained. In order to find the sample edge points, the center of the object was found by taking the average of all points in that blob. Taking the center of the blob as the center of a circle, rays are found even spaced from 0 to 360 degrees. The edge point that fell closest to each ray was chosen. This ensured the shape of the object is still represented with a reduced number of edge pixels.

The edges were found for the second reference image. Shape context and TPS were then used to determine how close the two reference images were. The number of bins in the histogram effects how well images match. Utilizing a lot of bins will not only increase computation time but also drive the matching cost up. Fewer bins may not give high enough resolution between points. The number of bins making up the histograms can be determined by the number of radial divisions and the number of circles. The number of radial divisions and the number of circles were both set to 10, as specified by the user.

Two additional user defined variables are needed in order to compare images using shape context and TPS. A maximum cost for matching points should be defined when applying the Hungarian method. This allowed some points to remain unmatched rather than forcing a match and inflating the total cost. A necessary condition to complete the analysis is that at least one point had to match between the images being compared. The second variable is the weighting factor in the cost function between points, see equation 1. A weighing factor of 0.4 was used in this case. This gave the difference between histograms a bit more weight than the angle of the point.

The TPS method was then used to calculate the bending energy and to morph the second reference image to the first. A quick comparison between the resulting image and the first reference was done. The cost of matching points, the bending energy and the comparison between the morphed image and the reference were calculated.

Applying shape context and TPS to the two reference images gives a matching cost, a bending energy, and a measure of how similar the resulting morph is to the first reference. These values can then be used as thresholds. An object in the test images will be declared a Yellow Tang fish if it falls below these values.

IV. Results and Comparison

AVATAR, using the aspect ratio method, was compared to the shape context and TPS method. Each method used the same reference images in order to create ranges in which to recognize the object. These recognition thresholds were used to recognize the Yellow Tang fish in the test images.

Both methods were able to correctly identify all the Yellow Tang fish in the test images. In the final image, AVATAR incorrectly identified the tail of the yellow fish as a Yellow Tang fish. The method of shape context did not and was able to avoid the false positive. Both methods correctly identified the fish in the bottom left image as not a Yellow Tang. The Shape context and TPS method had fewer false positives than the aspect ratio used by AVATAR.

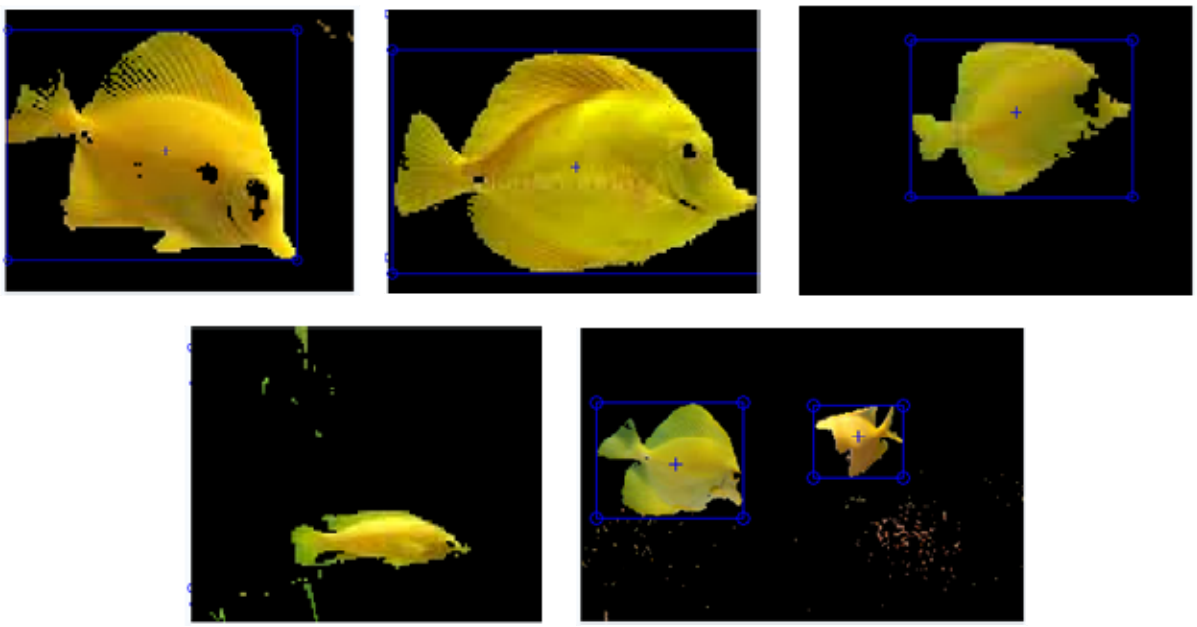

Figure 5: Results from AVATAR on Yellow Tang images. The far left image is the test image and therefore was used to determine the criteria for selection.

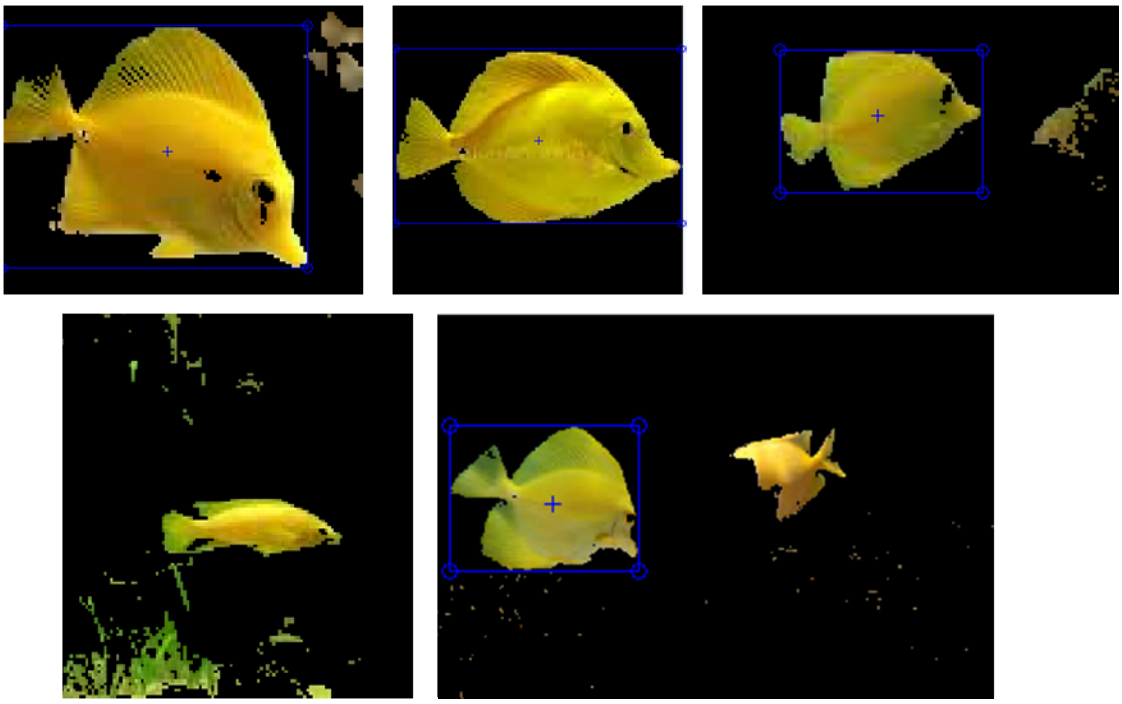

Figure 6: Results from the shape context and TPS method.

An autonomous sampling mission will most likely have the vehicle looking for multiple objects to sample. Therefore, the data for multiple reference objects must be saved. AVATAR is a much less computationally intense than shape context and TPS and much less data needs to be stored for each reference object. AVATAR only requires 12 values to identify if an object is of the same type as the reference or not. These 12 values are: minimum and maximum values for HSV color space, minimum and maximum aspect ratio, minimum and maximum area and minimum and maximum density ratio. In

order to compare a potential object to the reference using shape context the edge points of the reference object must be saved, in addition to the color ranges, minimum and maximum costs, as well as a few user specified terms. This could be hundreds of additional points.

Shape context is not able to correctly recognize images that have been mirrored. The histograms for the sample points between the objects are very different. As a result the method is unable to correctly match points between the reference image and the object it is trying to recognize. This can be mitigated by having a second reference image. However, this does require more information to be stored and doubles the amount of reference images.

V. Future Work

A. Sensitivity of Variables

There are many user specified variables in the shape context and TPS method such as the number of histogram bins, maximum cost between points, the weighing factor, and the number of sample points used. These variables can greatly affect the results. The number of sample points, for example, must be high enough to maintain the object's shape but not too high as more points can greatly slow down the recognition processes. In the same fashion, having too many histogram bins can not only slow the program but also artificially drive up the cost as histograms no longer match. This will result in poor point matching between the reference and potential objects. The sensitivity to these variables should be examined.

B. Reference Image

Both methods use a reference image. The selection of the reference image is generally not a concern as often only one or two images of the reference are available. However, a survey from an AUV, which is the precursor to a sampling mission, will mostly likely result in multiple images of a reference object. The question then becomes which image to choose.

It was noted that while investigating the use of shape context and AVATAR, the reference image is extremely important. Some images match better to the reference than others. Given multiple images, the reference image could be the image that can be found to minimize false positives and maximize correct recognition with objects in a training set. An 'average' shape could be constructed from various target images in order to create an 'average' reference.

C. SAMURAI Implementation

SAMURAI is capable of autonomously recognizing and grabbing objects specified by AVATAR. Shape context and TPS should be implemented onto the SAMUAI arm. This will allow testing to be done to determine how well this method can be used to perform autonomous sampling. The code is being integration with SAMURAI.

In order for a manipulator to autonomously sample an object, the type of object must be known and the location must be known. Shape context and AVATAR both focus on how to recognizing objects. The method of determining the location of the object lies in the stereovision system. There are two cameras attached to SAMURAI, placed about 2 inches apart. Knowledge of the distance between the cameras allows the distance of an object to be determined when found in both frames. The first step, after calibration which was done using CalTech's MatLab Calibration toolbox, is to recognize an object in both cameras frames. Once the object has been recognized in both, then the distance to the object can be found using calibration data. [8]

Autonomous sampling is done in the real world where objects have 3D pose rather than just 2D. Most of AVATAR's testing was done with rubber ducks from a distance – which often reduced the duck shapes to more or less spheres or ellipsoids. As a result the ducks had the same shape from most points of view. Both methods are capable of finding a 3D object by using multiple reference images. For example with a complex object with different shapes depending on the view AVATAR would need multiple reference images of the complex object. Using multiple reference images at different points of view will still allow AVATAR to recognize the complex object but at a computation cost. Shape Context can be used the same way; however, Shape Context can be used to determine the 3D pose of an object with only few additional reference images. It has been shown that Shape Context can correctly recognize 3D objects with only a few views [2].

VI. Conclusions

Autonomous sampling is a very complex problem. A vision system must be able to recognize a 3D object in the real world and determine the distance so that a manipulator can perform the sampling. AVATAR currently uses a comparison of aspect ratios to recognize objects. Shape context and TPS use the entire shape of the object for recognition. The shape context and TPS method results in fewer false positives but is more computationally complex. Shape context can be used by an autonomous sampler in order to sample specified objects types.

VII. Acknowledgements

The authors would like to thank our sponsors throughout our development of these systems: David Lavery and the National Aeronautics and Space Administration, work conducted under grant NNG04GB31G; Kandice Brinkley and the National Science Foundation, work continues to be conducted under grant OCE0752347; and the Maryland Industrial Partnerships program, under grant 4211

The authors would also like to thank Hanumant Singh and Robert Sohn of the Woods Hole Oceanographic Institute for their collaboration, help and use of the JAGUAR AUV.

VIII. Works Cited

[1] M. P. Belongie, "Shape contex: A new descriptor for shape matching and object recognition," *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS,* pp. 831-837, 2000.

[2] S. J. M. Belongie, "Matching with Shape Context," in *Content-based Access of Image and Video Libraries, 2000. Proceedings. IEEE Workshop on* , 2000.

[3] M. Naylor, Autonomous Target Recognition and Localization for

Manipulator Sampling Tasks, College Park: University of Maryland, 2006.

[4] C. Lewandowski, D. Akin, B. Dillow, N. Limparis, C. Carignan, H. Singh and R. Sohn, "Development of a Deep-Sea Robotic Manipulator for Autonomous Sampling and Retrieval," in *Autonomous Underwater Vehicles 2008*, 2008.

[5] S. Belongie, J. Malik and J. Puzicha, " Shape matching and object recognition using shape contexts," in *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , 2002.

[6] F. Bookstein, "Principal warps: thin-plate splines and the decomposition of deformations," *Pattern Analysis and Machine Intelligence, IEEE Transactions on ,* vol. 11, no. 6, pp. 567 - 585 , 1989.

[7] D. A. C. C. Barrett Dillow, "Development and Testing of a Dexterous Manipulation Capability for Autonomous Undersea Vehicles," in *AIAA Infotech@Aerospace Conference and AIAA Unmanned/Unlimited Conference*, Seattle, 2009.

[8] J.-Y. Bouguet, "Camera Calibration Toolbox for Matlab," CalTech, 9 July 2012. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/.