An automatic identification and monitoring system for coral reef fish

Joseph Wilder[a], Chetan Tonde[a], Ganesh Sundar[a], Ning Huang[a], Lev Barinov[a], Jigesh Baxi[a], James Bibby[a], Andrew Rapport[a], Edward Pavoni[a], Serena Tsang[a], Eri Garcia[b], Felix Mateo[b], Tanya M. Lubansky[c], Gareth J. Russell[c]

[a]Rutgers, The State Univ. of New Jersey; [b]Union City High School; [c]New Jersey Institute of Technology

ABSTRACT

To help gauge the health of coral reef ecosystems, we developed a prototype of an underwater camera module to automatically census reef fish populations. Recognition challenges include pose and lighting variations, complicated backgrounds, within-species color variations and within-family similarities among species. An open frame holds two cameras, LED lights, and two 'background' panels in an L-shaped configuration. High-resolution cameras send sequences of 300 synchronized image pairs at 10 fps to an on-shore PC. Approximately 200 sequences containing fish were recorded at the New York Aquarium's Glover's Reef exhibit. These contained eight 'common' species with 85–672 images, and eight 'rare' species with 5–27 images that were grouped into an 'unknown/rare' category for classification. Image pre-processing included background modeling and subtraction, and tracking of fish across frames for depth estimation, pose correction, scaling, and disambiguation of overlapping fish. Shape features were obtained from PCA analysis of perimeter points, color features from opponent color histograms, and 'banding' features from DCT of vertical projections. Images were classified to species using feedforward neural networks arranged in a three-level hierarchy in which errors remaining after each level are targeted by networks in the level below. Networks were trained and tested on independent image sets. Overall accuracy of species-specific identifications typically exceeded 96% across multiple training runs. A seaworthy version of our system will allow for population censuses with high temporal resolution, and therefore improved statistical power to detect trends. A network of such devices could provide an 'early warning system' for coral ecosystem collapse.

Keywords: Coral reefs, fish populations, population census, image processing, pattern recognition, species recognition

1. INTRODUCTION

Coral reefs and the complex communities they support are in decline worldwide[1], and while the causes are many and complex[2], the role of climate change is no longer in doubt. Both increases in surface ocean temperature and the resulting acidification caused by greater direct absorption of carbon dioxide have been implicated in rapid coral die-off, or 'bleaching,' in which the coral hosts lose their symbiotic algae[3,4]. A key unknown is whether reef-dwelling invertebrate and fish communities will be able to adapt to these often dramatic changes, either in the short term (if coral die-off is temporary) or the long term (if it is frequent and periodic, or permanent), and what form such adaptation will take.

By far the limiting factor in improving our understanding of reef community responses to disturbance (and reef community dynamics in general) is the difficulty of data collection. Counting all individuals of a set of different species is intensive and time-consuming. The problem is magnified enormously by the need to be underwater. In practice, censuses of shallow-water marine life are performed by divers using visual identification[5]. The short time period allowed in scuba dives, along with logistical complications and expenses (the need for training, equipment, boat transport, etc.) and risk (open-water diving is inherently risky compared to most research endeavors) mean that censuses are few and far between. A typical study will be based on data from short visual surveys (10–20 minutes is common), at a small number of locations, occurring perhaps once a month for a year or two[6].

If there were a means to collect high-quality data almost continuously, analogous to the way one might collect seawater temperatures using a permanent monitoring station, we would then have something approaching an early warning system. This project aims to develop just such an instrument. Our short-term goal is to develop a proof-of-concept prototype of an underwater sensor module to automatically census coral reef fish populations. Our ultimate goal is to

Applications of Digital Image Processing XXXV, edited by Andrew G. Tescher, Proc. of SPIE
Vol. 8499, 84991H · © 2012 SPIE · CCC code: 0277-786/12/$18 · doi: 10.1117/12.928860

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

develop a rugged, sea-worthy version of this module that can be replicated at a number of locations around a coral reef that will send data to a central remote location where long-term trends in fish population rises and declines can be calculated. This system can, hopefully, serve as the proverbial "canary in the mine shaft" for the health of coral reefs.

The following sections outline the hardware and software development involved in creating our prototype, the fish populations we encountered in our initial installation at the New York Aquarium, the feature extraction and recognition techniques employed, the results of our preliminary pattern recognition analysis for species identification, and the next steps in the development of a fully operating system.

2. HARDWARE DEVELOPMENT

Hardware development was carried out in two phases. During the first phase, a small aquarium was built, configured with two cameras and appropriate illumination, and stocked with a variety of coral reef fish. This phase enabled us to configure our cameras for image acquisition, develop an operator interface and carry out basic image processing operations.

Phase two involved the design, construction and testing of a fully submersible prototype system that could be employed in an underwater environment that closely matched that which would be encountered in an actual coral reef. This system was installed and tested in the coral reef exhibit at the Wildlife Conservation Society's New York Aquarium. This exhibit is a model of Glover's Reef in Belize, the location of the Society's Glover's Reef Research Station.

2.1 Original concept

Fig. 1 shows the original concept sketch. The final instrument followed the concept in general outline, the main difference being the nature of the camera housings (which were hemispherical dome-type) and lights (which were mounted independently to the frame — not shown in the sketch).

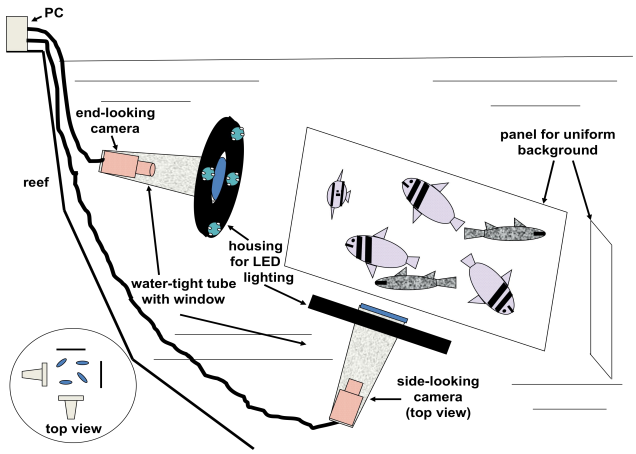


Fig. 1. Original concept sketch.

2.2 Construction of the prototype: Phase 1

In phase 1 of the project we set up the camera-computer interface for image acquisition by mounting the cameras externally to a small aquarium. A suitable aquarium was designed and built, with help from the Rutgers University Institute for Marine and Coastal Science (IMCS). It was configured as a cube, holding approximately one cubic yard of water. It had two transparent adjacent sides and two sides with matt black panels, simulating the design for the coral reef setup (see Fig. 1). A flexible image capture system was developed that captures simultaneous sequences using two, compact, high-resolution GIG-E cameras that support up to 100m cable lengths between cameras and an on-shore PC. The 1360 x 1024 pixel Prosilica GIGE GC1380C cameras support multiple programmable functions, e.g., frame rate, integration time, gain, and region of interest, to best interact with a difficult and constantly varying environment that will be encountered in the deployable prototype. The cameras were connected to a powerful PC via high-speed GIG-E data acquisition cards. Eight high-powered, compact LED modules were acquired to provide illumination throughout the tank. Using diffusing plates and mirrors, we constructed a system that distributed the illumination uniformly throughout the tank, using a compact mounting system. A pumping system was installed that kept fresh seawater flowing through

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

the tank to duplicate, as closely as possible, the environment we expected to encounter along the coral reefs, where the system will be installed, ultimately.

Prior to stocking the small aquarium with fish, the existing high-intensity LED illumination was augmented with overhead fluorescent illumination to simulate sunlight conditions that would be found during daylight hours on a coral reef. Four fluorescent fixtures with bulbs of color temperature approximating sunlight were mounted above the tank (see Fig. 2). A variety of species found along coral reefs were added. Images from the two cameras are shown in Fig. 3.

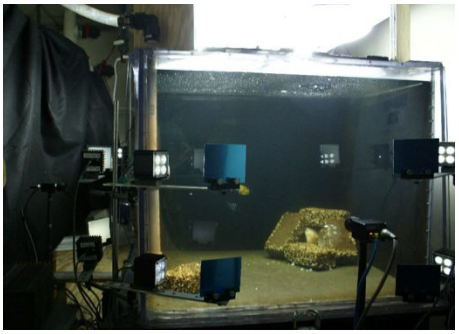


Fig. 2. Small tank with cameras and illumination system.

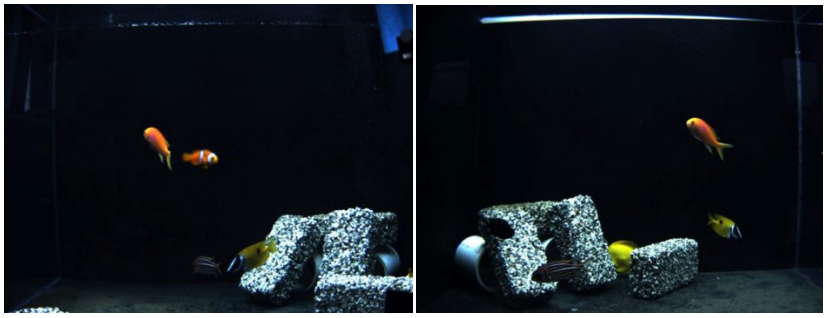


Fig. 3. Simultaneous coral reef fish images from cameras 1 and 2.

2.3 Construction of the prototype: Phase 2

The initial concept for the prototype (see Fig. 1) was an open frame that holds two cameras, various lights, and two 'background' panels that block out the complicated scenes of more distant regions (Fig. 4a). The idea was to study smaller species (less than 20 inches in length) that live close to coral reefs to seek refuge from larger predators.

One challenge in designing the system was to balance two conflicting requirements. The first is the need to keep the camera housings as close as possible to the frame within which the fish are to be viewed for reasons of structural rigidity. The second is the need to choose a lens with a sufficiently wide field of view to be able to view as much of the interior of the frame as possible without imaging regions far outside of the frame, which results in loss of resolution where you need it. The compromise lens selection and distance from the frame was an 8.5 mm lens at a distance of 18 inches from the frame. This selection meant that each camera views a portion of the interior of the frame within a centered, truncated pyramid whose cross sections consist of rectangles with width/height ratios in the proportion 1360:1024. The portion of the interior of the frame visible to both cameras was the intersection of these two pyramids at right angles to each other, and constituted the majority of the volume of the frame. The regions outside of this large central volume consisted of corners where a fish may appear in one of the cameras but not the other, and small regions not visible to either camera. We found it necessary to insert an 8 inch high platform in the bottom of the frame to prevent bottom-feeding fish from swimming entirely underneath our viewing volume.

Two other issues had to be considered as a result of choosing a wide-angle lens. The first one was a three-fold change in magnification between the nearest distance (18 inches) and the farthest distance (58 inches) from each camera. To compensate, we created a calibration map of all of the voxels in the image using checkerboard pattern. The second issue

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

was image distortion due to the wide angle, but this turned out to be insignificant for the portions of the images in the interior of the frame.

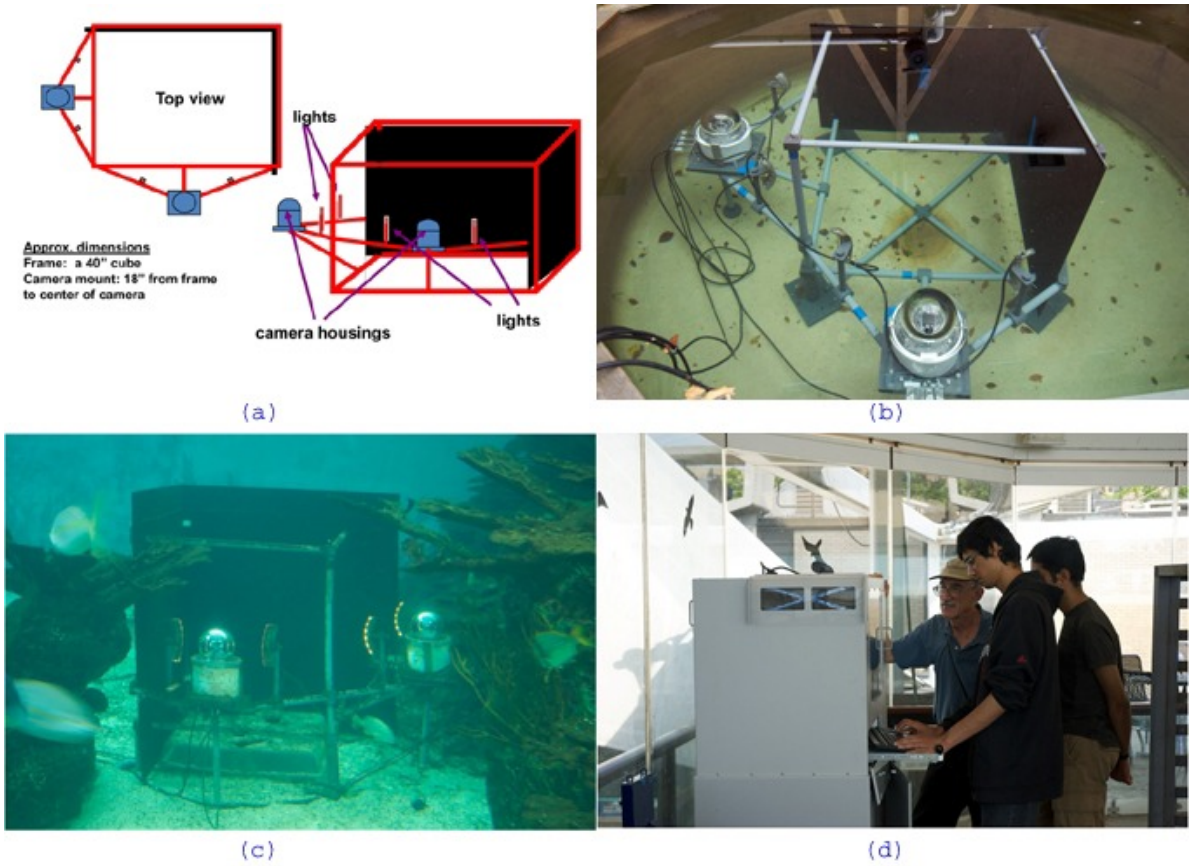


Fig. 4. (a) Frame concept design. (b) Initial test of submersible system. (c) System operating at New York Aquarium coral reef exhibit. (d) System instrumentation enclosure.

The cameras, described above in the Phase 1 section, were installed in secure underwater housings that operate at the required depths for our target reefs (from Ocean Presence Technologies). They are waterproof to a depth of 30m and provide minimal image distortion via a hemispherical "dome" viewport. Underwater illumination modules were also acquired from Ocean Presence Technologies. The prototype was constructed of high-density PVC tubing and brackets and polycarbonate panels, which are attached to each other with a combination of marine glue and stainless steel screws. The polycarbonate panels are black throughout, and were sanded with ultra-fine-grit sandpaper and painted with ultra-flat camouflage paint to produce a matte finish on their interior-facing surfaces. When wet, this produces a very uniform, light-absorbing surface. These construction materials were judged to be an appropriate compromise between cost/weight and strength/rigidity for the prototype instrument. The frame was designed to accept the weight of the chosen camera housings. The cameras, described above, provide suitably high frame rates (e.g., 10fps) over long cable runs with a relatively low power requirement. The completed prototype was tested in a tank at the Rutgers Institute of Marine and Coastal Sciences (Fig. 4b), and installed at the New York Aquarium coral reef exhibit (Fig. 4c), along with its environmentally secured instrumentation enclosure from Dustshield (Fig. 4d).

3. SOFTWARE DEVELOPMENT

The block diagram in Fig. 5 shows the image capture, image processing, feature extraction and pattern recognition stages of our instrument. The details of each stage are presented in the following paragraphs.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

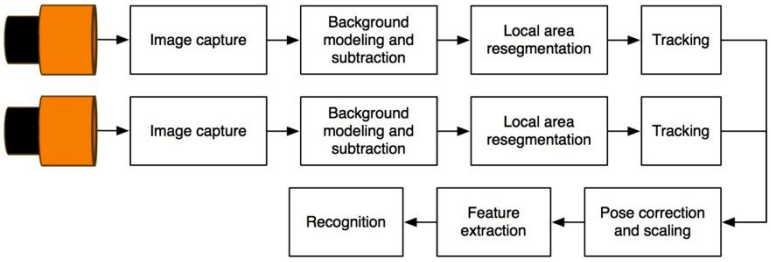


Fig. 5. Image capture, processing, feature extraction and pattern recognition flowchart.

3.1 Image capture

Figure 6 shows the flow involved in the capture of image sequences using Prosilica GigE interface cameras. Two independent threads configure each camera independently and simultaneously transfer images from the interface to RAM which subsequently is written to disk.

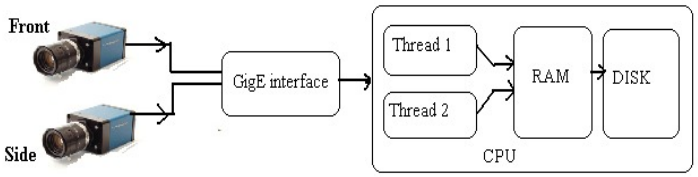


Fig. 6. Image capture flow diagram.

Prosilica cameras support fixed frame rate capture and we tested frame rates of 5, 10, 15 and 20 fps, settling on 10 fps for most data acquistition. The capture sequence is triggered simultaneously in both cameras, which start capturing the frames independently. Capture time, frame rate and image pair time delay are computed using the computers internal system clock up to an accuracy of +/–1 ms. The captured frames from the cameras are transferred to the GigE interface which stacks up to 100 frames in a queue. Images from the queue are simultaneously transferred from the queue to internal RAM storage as uncompressed TIFF files. All the captured images corresponding to each camera are then written to a secondary storage, i.e., hard disk, for training and testing purposes.

3.2 Control software

A Graphical User Interface was developed that allows a user to interactively align the $\{x, y\}$ coordinates of the two cameras, initiate a capture sequence, followed automatically by the preliminary processing steps of background subtraction, segmentation and object tracking. (This interface is useful for testing — when fully operational, the system is triggered automatically to collect and process image sequences on a pre-defined schedule.) The various steps involved in using the GUI are as follows:

X/Y pixel alignment. Both cameras capture the images of two white rectangular patterns at the top and bottom of the intersection of the two black background panels. The centroids of the two patterns are computed in each camera's image and the midpoint of the line joining the centroids is used to make software adjustments to align the x/y coordinates in both cameras.

Image capture. An image sequence capture is programmed by selecting the number of frames in the sequence and the frame rate. In our experiments we captured image pairs at a rate of 10 per second for 30 seconds, yielding 300 image pairs. These are then written to the computer's hard drive.

Background subtraction and segmentation. After dividing the image sequence frames into two parts, a training sequence for modeling the background and a testing sequence for segmenting the image into foreground (the fish) and the background (the background panels and other non-fish items), each frame was processed to extract rectangular regions that are candidate fish images. A re-segmentation step to refine the original segmentation is also included here.

Tracking. Following a fish during its trip through the viewing volume and disambiguating the tracks of multiple overlapping fish is carried out during this step.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

The details involved in these pre-processing steps, as well as other image processing and pattern recognition operations leading to the identification of coral reef fish, are outlined in the following sections.

3.3 Camera calibration

Understanding the geometry of the imaging system requires knowledge of the intrinsic and extrinsic parameters of the cameras. The intrinsic parameters of the cameras are fixed and depend on the design of the camera sensors and their focus systems. The extrinsic parameters characterize the 3D location of the cameras with respect to the world reference frame. The intrinsic and extrinsic matrices together describe the complete image formation process from a 3D point in the world to a pixel location in the image. They provided the data for calibration of the cameras computed from checkerboard images taken at different distances and orientations when the system was installed underwater at the New York Aquarium.

3.4 Background modeling and subtraction

Background subtraction is needed to extract foreground features of interest for further processing. The static background includes the background panels (present in both our original tank and the fully submersible system) and any rocks or other permanent features that contain little information about the fish themselves. By removing the static background we can find out where the fish are and perform further processing, e.g., tracking, pose correction and recognition, more easily. Ideally, we want the background subtraction stage to also remove small floating particles and ignore the algae growth that we found during the period when our system spent several months underwater at the New York Aquarium.

A two-step process was developed two carry out the background modeling and subtraction, the first operating on the entire field of view and a second focused on refining the process in the immediate neighborhood of fish found during the first step. This two-step process worked very well, even as the black panels became increasingly covered in algae growth (this introduces fine-grained noise into the background-subtracted image that is easy to filter out). The first step, itself, concatenates two algorithms, a Kernel Density Estimation (KDE) method proposed by Elgammal et al[7] followed by a Graph Cut algorithm, an energy minimization method to solve Markov Random Field (MRF) problems in computer vision, Boykov et al[8]. The first algorithm involver kernel density estimation over *n* recent values of a pixel. It uses normal kernel smoothing over the histogram of the *n* most recent values of a pixel along with a neighborhood- and connected component-based conditional probability model. (This algorithm, compared with individual pixel-only algorithms, e.g. Stauffer and Grimson[9], was less sensitive to outliers from the recent past and was better able to distinguish pixels on a fish with colors and intensities similar to that of the background.) The approach consists of three stages; pixel displacement probability computation, connected displacement probability computation, and shadow removal. Each stage is explained below.

Pixel displacement probability. The pixel displacement probability is defined as

$$P_N = \max_{y \in N(x)} Pr\left(x_t | B_y\right),$$

where P_N is the maximum probability that the observed value, x_t, belongs to the background distribution of some point in the neighborhood $N(x)$ of x, and B_y is the background sample for pixel y. Probability estimation is carried out by using a Normal kernel density estimation window given by

$$P_N(x) = \frac{1}{N} \sum_{i}^{N} \prod_{d=1}^{3} \frac{1}{\sqrt{2\pi\sigma_i}} e^{\frac{1}{2}\left(\frac{x - x_i}{\sigma_d}\right)^2}$$

The kernel bandwidth chosen here is diagonal, with each σ_j corresponding to independent color channel variance. The variance is chosen as $\sigma_j = m/\left(0.68\sqrt{2}\right)$ where m is the median of the absolute difference values. This helps the model suppress small motions in the background, but also causes more false detection when foreground colors are similar to the background.

Connected component probability: To suppress false detection, an additional connected component-based conditional probability is computed. It is the probability that the pixel moved is from a component C from a nearby location given by

$$P_C = \prod_{x \in C} P_N(x)$$

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

This probability will be small if the component displaced from the background is from a real target in the foreground. Combining $P_N(x)$ and $P_C(x)$ we get the result that a pixel is background if and only if $(P_N(x) > th1) \wedge (P_N(x) > th2)$ is true. The background model is then successively repeated either selectively, based on change if there are pixel values in the derived mask, or blindly. Long term or short term models are considered for model updating.

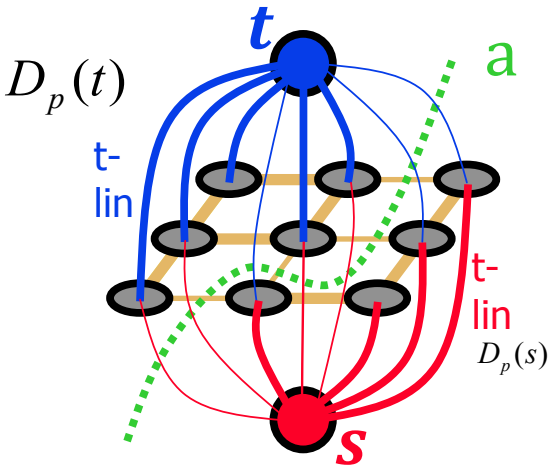
Shadow removal: The probabilities based on $P_N(x)$ and $P_C(x)$ do not handle shadow removal very well and they need to be treated separately. For this purpose, chromatic coordinates are used:

$$r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}, s = R+G+B$$

Density estimation over the (r, g) color space and simple thresholding removes shadows while still detecting the object. For more details of the shadow removal process see Elgammal[7].

Although this algorithm performed better than pixel-only algorithms, it still had some difficulty in creating complete binary masks when portions of a fish image were similar in color and intensity to that of the background.

The second concatenated algorithm is an energy minimization method called *graph cut.* The figure below shows a graph whose nodes are connected by weighted edges, and also connected to the source *s* and the sink *t*. We are trying to find a minimum cut to divide the graph into source *s* and sink *t*. Ideally, a cut through the least weighted edges (as shown in the graph) gives the bipartition of the graph with minimum cut. In our application, the weights between each pixel and the source are probabilities of being background, edge weights between pixels are given by color similarities.



The energy we are trying to minimize is in the form of

$$E(f) = \sum_p D_p(f_p) + \sum_{p,q \in N} V_{p,q}(f_p, f_q)$$

where f is the labeling at a given node, $D_p(f_p)$ assigns a data term for the given labeling, and $V_{pq} \cdot (f_p, f_q)$ is the smoothness term between neighboring pixels p and q. The data term comes from the result of background subtraction using Kernel Density Estimation (KDE), and the smoothness term comes from the pixel similarities between neighbors.

The pixel-based background subtraction algorithms did not take the spatial relationship between neighboring pixels into account. The graph cut algorithm with the smoothness energy solves this problem by assigning neighboring pixels with similar colors to be the same (either foreground or background).

Finally, we eliminate blobs that are < 0.01% and > 50% of the image area. The second step is a local 'clean-up' re-segmentation using the same graph cut algorithm but applied only to the areas within any bounding boxes extracted in the previous stage (Fig. 7). The end result is illustrated in Fig. 8 (A) and (B), showing an image pair captured at the New York Aquarium.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

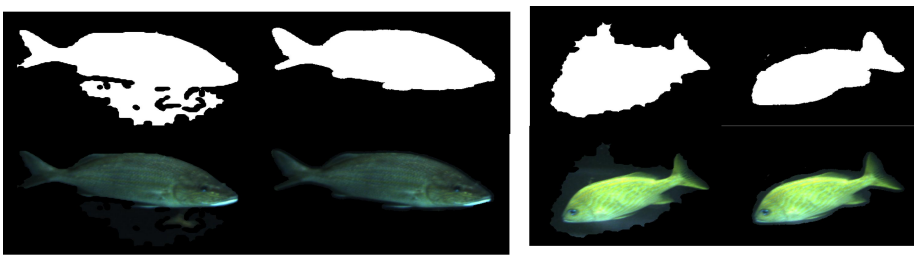


Fig 7. Two examples of re-segmentation using immediate image neighborhood.



Fig 8. Segmentation of fish from their background (A, B) and tracking (C). In (C), solid box denotes foreground detection from background modeling, dashed box denotes the tracker, number indicates the object id.

3.5 Tracking

A particle filter-based tracking algorithm is used to track fish across frames and also to detect overlaps between multiple fish (Fig. 8C). A very brief summary of the procedure is as follows (for more details see the MS thesis of Chetan Tonde[10] entitled "System Integration and Image Pre-processing for an Automated Real-Time Identification and Monitoring System for Coral Reef Fish", Rutgers University, 2010):

1. Locate the object in the first frame from background modeling; represent it as a single particle $X_t = [x_t, y_t, \theta_t, s_t, \alpha_t, \varphi_t]$ in the affine space (where x_t is the x translation, y_t is the y translation, θ_t is the rotation angle, s_t is the scale, α_t is the aspect ratio and φ_t is the skew direction).
2. Initialize the eigenbasis to be empty and the mean image to the appearance model in the first frame.
3. Move to the next frame and draw particles from the particle filter according to the dynamical models using the affine parameter variances

$$p(X_t|X_{t-1}) = N(X_t; X_{t-1}, \Psi)$$

4. For each particle, extract the corresponding image patch in the current frame and calculate its likelihood:

$$p(I_t|X_t) = p_{d_t}(I_t|X_t)p_{d_w}(I_t|X_t) = N(I_t; \mu, UU^T + \epsilon I)N\left(I_t; \mu, U\sum x^{-2}U^T\right)$$

5. Select the most likely particle and store the corresponding image patch.
6. After collecting desired number of new images, update the mean image with a forgetting fact (f) using equation below:

$$I_C = \frac{fn}{(fn+m)}I_A + \frac{m}{(fn+m)}I_B$$

7. Go to step 3 and continue, repeating the above process until the object exits the field of view or is lost for 7 successive frames, as observed from the background subtraction algorithm.

3.6 Pose correction and scaling

This was carried out via epipolar analysis, using information from both cameras to provide depth information, pose correction and disambiguation of overlapping fish. Note that 3D stereo-based reconstruction, which may prove difficult in this application, was *not* employed here. The depth information *was* used for scaling and to aid in disambiguation of overlapping fish. Also, having two simultaneous views of a fish in a difficult pose situation affords the opportunity for selecting the better of the two views for further processing. Note that with our current configuration of components, there were a number of instances in which a fish passing through the system was visible in only one camera for some or all of

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

an image sequence, so that full epipolar analysis was not possible. (Our planned second-generation system will provide two simultaneous camera views for a much larger percentage of fish image sequences.) All extracted fish images were rotated so that a line from the mouth to the centroid if the body was horizontal, prior to proceeding with feature extraction and pattern recognition. Visual inspection suggested that this gave more consistency between images than rotation based on major axis, because it aligns the less flexible front portion of each fish.

4. THE FISH IMAGE DATABASE

There were 26 species of fish shorter than 20 inches in length (our criterion for small reef fish) in the New York Aquarium's Glover's Reef exhibit, 23 of which belonged to eight families and 3 of which were unique (see Fig. 9).

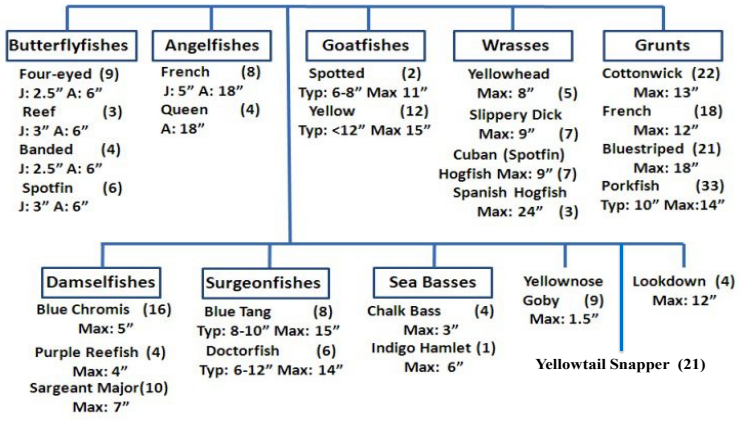


Fig. 9. 26 species, 23 of which belong to 8 families and 3 of which are unique (parentheses show how many are in the NY Aquarium exhibit).

Fish from the same species may have small differences in color and patterning, especially in mature vs. immature fish. Fish species from the same families are usually similar in shape, but may differ in coloring and internal patterns. Examples of challenging within-species and within-family variations in appearance can make accurate identification of fish difficult (see Fig. 10).

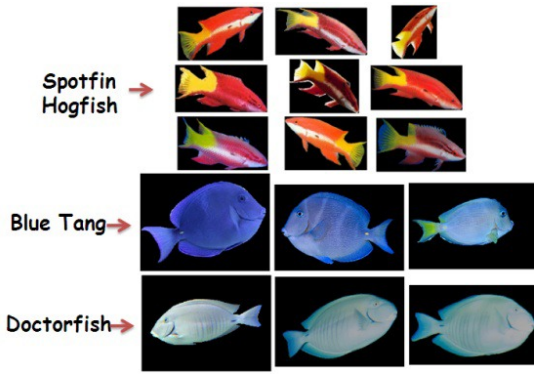


Fig. 10. Recognition challenges: within species (Spotfin Hogfish) and between species (Blue Tang and Doctorfish).

Approximately 200 sequences containing coral reef fish belonging to 16 of the 26 target species were recorded during initial tests at the New York Aquarium's Glover's Reef exhibit. (The other species did not enter the camera system when we were recording images.) We were able to capture and successfully segment a large number of images, under various pose and lighting conditions, of 8 of the 16 species (we will call them the "common" species). We were able to capture only a small number of images of the remaining 8 (we will call them "rare") species, which we considered to be too few in number for meaningful classification. The common and rare species are listed in Table 1. Our recognition experiments

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

involve classifying the "common" species and treating all of the" rare" species as "unknown", ascertaining the degree to which they are falsely recognized as one of the common species.

Table 1. Number of sample images of 8 "common" and 8 "rare" species. 10 images were impossible to classify definitively even 'by hand' because of poor pose or lighting.

Common Species	**Rare Species**
Four eyed Butterflyfish – 107	Blue Chromis – 7
Banded Butterflyfish – 92	Purple Reeffish – 5
Cuban Hogfish – 180	Sergeant Major – 20
French Grunt – 672	Goatfish – 13
Blue Tang – 85	Queen Angelfish – 27
Porkfish – 188	Spanish Hogfish – 17
Bluestriped Grunt – 641	Cottonwick Grunt – 6
Yellowtail Snapper – 88	Hogfish – 25
	Unknown species – 10

5. FEATURE EXTRACTION AND CLASSIFICATION

5.1 Methods

For image classification we use a combination of shape, color and 'banding' features. Shape features are derived from 50 equally spaced points around the perimeter of the fish. The 100 x and y coordinates are first reduced to the 16 largest principal components, which encompass almost all the variation in the raw data. Because we use separate, dynamically generated testing and training sets in our classification analysis (see below), this data reduction step is re-performed (on the training data) each time, and we save the linear coefficients from the principal component analysis so that we can apply the same transformation to the testing data.

Color features are two eight-bin histograms, one for each of the two 'color' channels after projecting the image from RGB into the Opponent color space: $o1 = (R - G)/\sqrt{2}$ and $o2 = (R + G - 2B)/\sqrt{6}$. Finally, 'banding' features are components 4 though 19 of a discrete cosine transform of a 256-element vertical projection of a grayscale version of the fish image. Thus we end up with 16 x 3 = 48 features per image.

For classification, we impose a minimum threshold on how many images we need for a species before we will try to classify it. Species with fewer images are lumped into a single, heterogeneous 'unknown' category. In our data, species divided neatly into two groups, which we labeled 'common' and 'rare'. The eight common species had between 85 and 641 images, and the rare species between five and 34 with a combined total of 154. Thus we had nine groups to classify, and each feature vector was labeled accordingly.

5.2 Neural network

For classification we used a feedforward (backpropagation) neural network architecture with sigmoid neuron functions and four hidden layer neurons, implemented in Mathematica 8 (Wolfram Research) using the add-on Neural Network application package. In our initial system, the network was trained for either 50 iterations (on the full data) or 25 iterations (whenever a subset of species was used). The networks were fit to training data and validated using independent testing data. To do this, we partitioned our image set into four independent subsets using stratified sampling (so that species would be represented approximately the same in each subset as in the whole dataset). We then repeated our training process four times, each time using one of the subsets as the testing data and the remaining three-quarters of the images as the training data. In our presentation of final results, such as confusion matrices, we aggregate the results of all four training/testing cycles, but the identities of the testing sets and the networks derived from them are maintained throughout the hierarchical development cycle described below.

5.3 A hierarchical classification system

We implemented the neural network-based classification in a hierarchical manner that mimics the way in which biologists manually classify species. In the first round of classification, data for all species are provided as input, and the

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

classifier attempts to assign them into one of nine classes (eight species and 'unknown'). Comparing the classifications of the testing set with their known identities generates a confusion matrix in which off-diagonal elements represent classification errors. The impetus for the hierarchical system is two observations. The first is that in most species classification systems, errors are concentrated in just a few locations. In other words, when species are misclassified, it is usually as just one or two of the other possibilities. (This arises naturally from the similarity structure of species sets, which in turn derives from their branching evolutionary history.) The second observation is that almost all classification systems perform better the fewer classes there are to separate. Thus we can implement another round of classifications in which we identify characteristic errors from the first found, and develop classifiers for each of these. In principle, these will be separating smaller numbers of species, and should perform better at this task than the full-system classifier.

Our confusion matrices have the true species ID on the rows and the assessed ID on the columns, so each column represents all the images that were classified as a particular species, and the off-diagonal elements in the column are those that were incorrectly classified as that species. These errors are what matter, because in a working system only the assessed ID is known. We therefore normalize the columns to percentages, and identify those elements in each column that are above an 'error threshold,' which we set to 2%. These indicate which species are usually mistaken for the species in question, and for which we will therefore develop a classification subsystem.

The confusion matrix has nine columns, and in our dataset usually more than half of them have errors above the threshold, leading to the development of subsystems that will be used if the first round classification is one of the relevant species. Subsystem development is essentially the same as for the main system (i.e., using the same neural network approach), except that the testing and training datasets are subsampled to include only the relevant species for each subsystem, and the number of network training iterations reduced.

After round two, we have a hierarchical classifier consisting of a single trained network architecture at the first level, and a set of up to nine networks in the second layer, corresponding to each of the possible outputs from the first layer. (If no second level network is present for a given first level output, then the first-level classification stands.) As well as the networks themselves, we store the associated shape PCA coefficients so that an input vector can be processed appropriately before being submitted to the network.

Repeated testing using a two-level hierarchical system showed that the final confusion matrices, while much improved over the single-classifier system, often had a small number of remaining error clusters (again using a threshold of 2%), almost always between two species. In that case we repeated the process of generating another level of networks, to produce a three-level final system. Fig. 11 shows a representation of a trained system.

The final system is a network of neural networks, and it is tempting to call it a meta-network, but we note that the implementation of a hierarchical system like this is independent of the nature of the classification algorithm itself. The only requirement is that the algorithm adapt and improve its performance with the reduction in the number of classes that occurs at each level, and that will be true of most algorithms (nearest neighbor, linear discriminant analysis, support vector machine, etc.).

As described earlier, in developing each hierarchical system we maintained the integrity of our testing and training sets throughout, so in fact for any given partitioning of our data into four testing sets, we end up with four trained hierarchical systems. We can assess overall performance after the fact by looping through all our images, sending each one to the appropriate system depending on which testing set it was originally from, and then aggregating the results into a single confusion matrix. A trained system generates three IDs, from the first, second and third levels, so that we can track how classification improves with the addition of the error-fixing subsystems.

There is one exception to the complete separation of the four testing/training combinations, which is the decision about which subsystems to develop. The confusion matrix generated from one of the combinations is based on only a quarter of the total data, and therefore is subject to additional stochastic variation based on the subsampling process. We wanted to have an idea about which errors are typical across *all* the data, and therefore we used the *combined* confusion matrix after level one to decide which subsystems should be developed for *all* four testing/training combinations (and the same after level two).

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

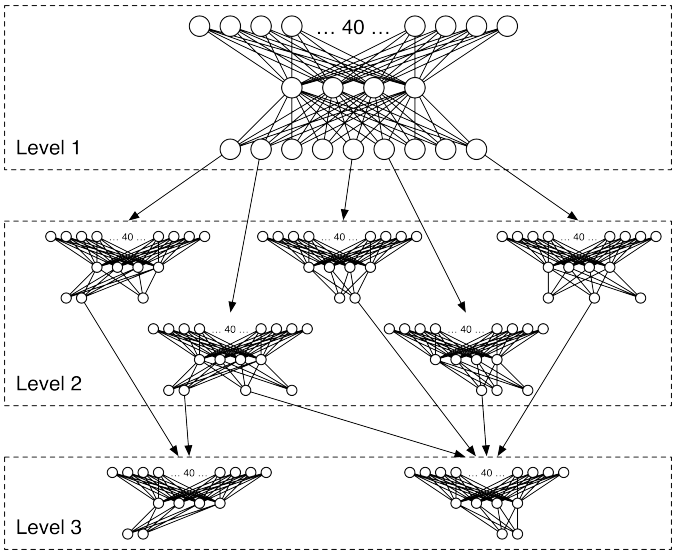


Fig. 11. An example of a neural-network based hierarchical classification system. The network in level one classifies a 42-element feature vector into one of nine classes (species in this case). Five of the classes (1, 2, 5, 6 and 9) cause the feature vector to be sent to a respective level two network, which has been trained to distinguish that species from those that the level one system commonly mistakes for it. Two of the possible class outputs from level two (2 and 6) lead to level three networks, which again attempt to fix any remaining errors.

6. RESULTS

Table 2 shows the combined testing-set confusion matrices following the first, second and third classification levels of one 'run' (the training of four separate systems based on four independent testing sets — see Methods). Rows indicate the true ID, and columns the assigned ID. Values are percentages by column. Bold elements show classification errors over the 2% threshold that leads to the creation of a subsystem on the level below. (One such error remains in the final matrix.) It is easy to see how the subsystems clean up errors. After the first round, performance for some species is excellent, but for others rather poor. After the level 2 systems are trained the performance is much improved, but there are still significant problems with Banded Butterflyfish IDs (column 2), and Blue Tang IDs (column 5), with a number of unknown images (and in the case of Banded Butterflyfish, Four-eye Butterflyfish images) sometimes being misclassified as one of those two species. After a third round of training in which those classifications are targeted specifically, accuracy of positive classifications is ~95% or better for all known species. Only the Unknown category has lower performance, but this is acceptable because in a working system, images classified as Unknown will be stored for later classification by hand. So a secondary goal for the system, after overall accuracy, is that when the system misclassifies one of the known species, it is into the Unknown class rather than into one of the other species classes. Serendipitously, the hierarchical process tends to achieve both goals. (As an example, in Table 2 if we compare level 2 with level 3, we see that the third level system changes errors in which Unknown images were classified as Blue Tang into more benign errors in which Blue Tang images were classified as Unknown.)

To assess whether the performance improvement seen in this run is typical, we performed several other runs. For each run a different stratified random partition of images into four independent testing sets is used. If improvement is seen across all runs, this eliminates the chance that it was caused by an unusual partitioning. In fact in all runs there was a substantial increase in accuracy after adding the second level, and for all but one there was a further small increase after adding the third level (Fig. 12).

7. CONCLUSIONS AND DISCUSSION

Our main conclusion is that an automated coral reef fish ID system is certainly feasible. Our recognition accuracy, based on a relatively generic set of features, was extremely high. This may seem unsurprising given the distinct visual presentations of most reef-dwelling fish, but it matches or exceeds that of experienced divers performing censuses (Lev Ginsberg, pers. comm.).

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

Table 2. Confusion matrices following each level in the hierarchical classification process. Each matrix is the combination of results from systems derived from four independent testing sets (see Methods). Values shown are *column* percentages.

LEVEL 1

Four-eyed butterflyfish	82.7	1.9							
Banded butterflyfish	**3.1**	70.5		0.2	**2.8**	**2.8**	0.2		**3.5**
Cuban hogfish		1.0	94.7						**1.7**
French grunt	0.8		0.5	96.6		1.1	0.2	2.2	
Blue tang	0.8	1.9	1.1		67.9				**5.2**
Porkfish	**4.7**	1.0		0.9	1.8	92.7	1.0	1.1	0.9
Bluestriped grunt		**14.3**	1.6	1.2	**7.3**	1.7	95.7	5.4	**14.8**
Yellowtail snapper	0.8		0.5	0.7				88.0	0.9
Unknown (Rare)	**7.1**	**9.5**	1.6	0.4	**20.2**	1.7	**3.0**	3.3	73.0

LEVEL 2

Four-eyed butterflyfish	98.1	**2.1**							
Banded butterflyfish	0.9	90.7		0.2					1.4
Cuban hogfish		1.0	94.7						1.4
French grunt			0.5	96.6		1.7	0.2		
Blue tang	0.9	1.0	1.1		92.9		0.3		0.7
Porkfish				0.9		95.6	1.4		1.4
Bluestriped grunt			1.6	1.2	1.2	1.7	97.2	1.2	0.7
Yellowtail snapper			0.5	0.7		0.6	0.2	98.8	
Unknown (Rare)		**5.2**	1.6	0.4	**6.0**	0.6	0.8		94.3

LEVEL 3

Four-eyed butterflyfish	97.2	1.1							
Banded butterflyfish	1.8	97.8		0.2					1.3
Cuban hogfish			94.7						1.9
French grunt			0.5	96.6		1.7	0.2		
Blue tang	0.9		1.1		97.3		0.3		**5.7**
Porkfish				0.9		95.6	1.4		1.3
Bluestriped grunt			1.6	1.2	1.4	1.7	97.2	1.2	0.6
Yellowtail snapper			0.5	0.7		0.6	0.2	98.8	
Unknown (Rare)		1.1	1.6	0.4	1.4	0.6	0.8		89.2

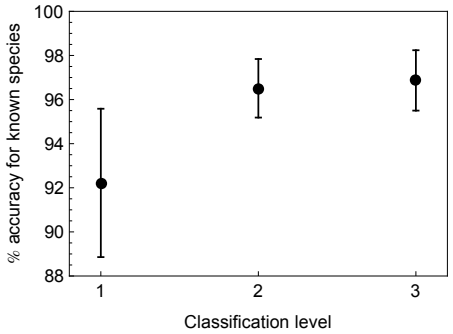


Fig. 12. Improvement of overall accuracy of 'known species' classifications with level in the classification hierarchy, across three training 'runs.' Each value is the sum of the diagonal elements of a confusion matrix divided by the grant total, after eliminating the final row and column that represent the heterogeneous 'unknown' set. (Each run involves a new partitioning of the images into four independent testing sets, the training of four separate systems based on these, and the aggregation of the final results.)

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

It remains to be seen how well this system will scale up to the challenge of distinguishing the larger numbers of species that will be found in a field site, but we note that many of the likely applications of a system like this do not require complete species coverage. For example, if the goal is to detect ecosystem shifts or other disturbances, close monitoring of the populations of a few of the most abundant species will provide more useful information than the occasional appearance (or otherwise) of the rarer species. Also, as has been emphasized in the development of other species ID applications, a key goal is simply to reduce the workload. In our system, 154 images were of rare or unknown species, and in a working system, images identified as unknown would be stored in an archive for an expert to identify, if possible, by hand. But this is 154 out of 2086, or > 8% of the total, and the fraction of *time* it would take to assess just the unknown images is trivial compared to the time required to directly census 2086 fish while diving.

It is likely that in a larger set of species, some will indeed be indistinguishable based on the features we used here, but there remains the possibility of developing targeted subsystems: those that use different, perhaps carefully chosen features to separate similar species. In fact we had begun this process with our system, before discovering that our three 'standard' feature types would perform so well. For example, the Porkfish has vertical stripes on its head but not the remainder of its body, and we had developed a separate feature set based on the DCT 'banding' signatures of the anterior, central and posterior 'thirds' of each image.

Having said that, a key usability advantage of maintaining the same feature set at all levels is that a hierarchical classifier can be generated in a completely automated manner (in about ten minutes for the systems in this paper). This potentially allows the system to be easily trained and used by those without advanced programming skills. In the context of a reef deployment, we envisage a cyclic training process. An initial set of images will have to be classified by hand, and these can then be used to generate the first ID system based on the more common species in the image set. Assuming the system works well, it will begin classifying the common species, and sending images of other species to an archive. An operator will periodically check the 'unknown' archive and identify the images species where possible. These will then be added to the training image set, and when a sufficient number had been added (for example, when another species has crossed the threshold image number for inclusion), a new ID system will be generated and tested (and compared to the existing system). In this manner the number of species that can be classified to species will increase the longer the system remains deployed.

Some further questions remain that will only be answered with data from a more species-rich, natural ecosystem, such as how the ID algorithms will cope with the gradual changes in morphology and color pattern that many species exhibit over the course of their life history, and whether these issues can be ameliorated (for example by careful siting of the physical components of the device so as to minimize visitations from juveniles). There is also the question of repeat visitation. It was certainly true in the New York Aquarium that some individual fish visited repeatedly, inflating the apparent count for their species. It is not clear how much this would happen in the wild, where the fish are not also constrained by glass walls to repeatedly encounter the camera system. (And while significant repeat visitation may rule out meaningful cross-species abundance comparisons, it will have a much smaller impact on the key goal of the detecting *within*-species temporal trends.)

In our view, the next main challenge is in obtaining good images from a field-sited system and sending them to a computer for the recognition steps. A reef-worthy system must be able to withstand at least moderate tidal flows without moving or bending. The components of our system most problematic in this regard are the background panels, against which moving water can generate substantial force. We may be able to improve matters with a mesh panel, or other form of venting. In any case, the system must be able to be easily lowered and fixed securely to an appropriate substrate (which may be coral itself, or an adjacent hard or sandy area). After assessing some of the physical weaknesses of our prototype, we envisage a "Mark II" system with simpler, lighter camera housings, a single background panel (which could be oriented parallel to the main tidal flow direction to reduce force), and a stainless steel or aluminum frame based on triangles for rigidity (Fig. 13).

Communications and power must also be considered. Our prototype was tethered, and while there are many reef environments within 100 m or so of land, that land may not itself be an ideal place to set up a large, power-hungry computer, or even to provide power. Ideally we would be able to use some forms of remote energy generation and wireless transmission, which in turn means we need to think carefully about how to reduce bandwidth requirements.

These are some standard problems in oceanic engineering and communications, and in many cases solutions already exist (for example, the manufacturer of our camera housings, Ocean Presence Technologies, offers a floating platform that combines wave-based power generation and wireless communication). We are optimistic that with advice and

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms

support from the right partners, a network of robust fish census stations could be deployed within a few years, monitoring our reef ecosystems and providing unprecedented data on fish population dynamics and distributions.

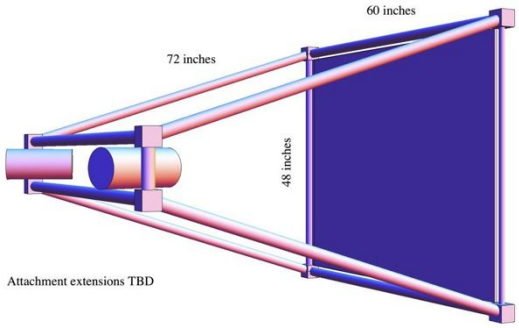


Fig. 13. Schematic for proposed "Mark II" instrument, showing two cameras viewing a single panel at a 45 degrees to one another.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (award #DBI-0649878). Felix Mateo and Eri Garcia were supported by the Research and Engineering Apprenticeship Program of the Academy of Applied Science.

REFERENCES

[1] Muscatine, L., Porter, J.W., "Reef Corals: Mutualistic Symbioses Adapted to Nutrient-Poor Environments". BioScience 27(7): 454–460 (1977).
[2] Wilkinson, C. (ed.), [Status of Coral Reefs of the World]. Australian Institute of Marine Science, (2002).
[3] Brown, B. E., "Coral bleaching: causes and consequences". Coral Reefs 16 Suppl.: S129–S138 (1997).
[4] Hoegh-Guldberg, O., P. J. Mumby, A. J. Hooten *et al.*. Coral reefs under rapid climate change and ocean acidification. *Science* 318: 1737–1742 (2007).
[5] Sale, P. F., Douglas, W.A., "Precision and accuracy of visual census technique for fish assemblages on coral patch reefs." Environmental Biology of Fishes 6(3–4): 333–339 (1981).
[6] Spalding, M. D., Jarvis, G.E... "The impact if the 1998 coral mortality on reef fish communities in the Seychelles", Marine Pollution Bulletin 44: 309–321 (2002).
[7] Elgammal, A., D. Harwood and L. S. Davis., "Non-parametric model for background subtraction." Proceedings of the IEEE 7th International Conference on Computer Vision (Frame Rate Workshop), Kerkyra, Greece, 1999.
[8] Boykov, Y., O. Veksler, Zabih,R.,. 2001. Fast Approximate Energy Minimization via Graph Cuts, International Journal of Computer Vision 11: 1222–1239 (2001).
[9] Stauffer, C., Grimson, W., "Learning Patterns of Activity using Real-time Tracking." IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8): 747–757 (2000).
[10] Tonde, C., "System Integration and Image Pre-processing for an Automated Real-Time Identification and Monitoring System for Coral Reef Fish", MS Thesis, Rutgers University, (2011).

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 10/17/2012 Terms of Use: http://spiedl.org/terms