Chapter 1

MULTI-LAYERED SCALLOP RECOGNITION FRAMEWORK

To identify *difficult-to-spot* objects like scallops from low resolution underwater images, specialized multi-layered process pipeline combining several computer vision and machine learning techniques can be developed; it also offers the flexibility to be easily retasked for other object detection domains.

Section 0: Intro

Section 1: A four-layered scallop recognition pipeline comprising a series of image processing, computer vision and machine learning techniques can be used to recognize scallop from noisy natural images.

Section 2: The first processing later, visual attention, hypothesized models of the human visual system and enables targeted search by identifying regions of interest in images.

Section 3: The second layer or segmentation layer bifurcates along these *interesting* regions obtained from visual attention into foreground and background pixels.

Section 4: The classification layer uses template matching to identify foreground objects that appear like scallops, with an emphasis on retaining true positives even at the cost of a large number of false positives.

Section 5: The final false positive filter layer uses another form of template matching to weed out the false positives from previous classification layer results.

Section 6: This scallop identification approach is unique in its ability to work on noisy underwater images and still offer *high* true positive detection rate.

Section 7: This multi-layered approach, specialized here to identify scallops, can be retrained with appropriate modifications to work with other objects or sea-animals.

1.1 Introduction

Recognizing marine organisms, like scallops, is a challenging problem. A previously introduced approach, named eigen-value based shape descriptors (in Chapter **??**), is incapable of utilizing textural information. Thus, eigen-value shape descriptors are unsuitable for recognizing organisms with prominent textural markers. Sensitivity to discretization noise, exhibited by Eigen-value based shape descriptors is another factor that discourages their use in noisy natural images. The multi-layered object recognition approach discussed in this chapter combines both shape and textural cues to recognize objects. This framework is also expressly designed to deal with noise present in images. A scallop enumeration problem is used as a means to validate this multi-layered approach.

The sea scallop *(Placopecten magellanicus)* fishery in the US EEZ (Exclusive Economic Zone) of the northwest Atlantic Ocean has been, and still is, one of the most valuable fisheries in the United States. Historically, the inshore sea scallop fishing grounds in the New York Bight, i.e., Montauk Point, New York to Cape May, New Jersey, have provided a substantial amount of scallops [1, 2, 3, 4, 5]. These mid-Atlantic Bight "open access" grounds are especially important, not only for vessels fishing in the day boat category, which are usually smaller vessels with limited range opportunities, but also all the vessels that want to fish in near-shore "open access" areas to save fuel.[1] These areas offer high fish densities, but are at times rapidly depleted due to overfishing [6].

The 2011 Research Set-Aside (RSA) project (Titled: "A Demonstration Sea Scallop Survey of the Federal Inshore Areas of the New York Bight using a Camera Mounted Autonomous Underwater Vehicle") was a scallop survey effort undertaken to study the health of the scallop population along the coast of New York-New Jersey. As a

[1] Based on personal communication with several limited access and day boat scallopers.

part of this effort around a quarter million images of the ocean floor were recorded and a manual scallop enumeration was performed on these images. The considerable human effort involved for manual enumeration spawned the idea of building an automated species recognition system that can sift through millions of images and perform species enumeration with minimal to no human intervention. In response to this need for an automated scallop enumeration system, a multi-layered scallop recognition framework was proposed [7, 8, 9]. The workflow of this scallop recognition framework involves 4 processing layers: customized Top-Down Visual Attention (TDVA) pre-processing, robust image segmentation, and object classification and false positive filtering layers.

The value of the proposed approach in this dissertation is primarily in providing a novel engineering solution to a real-world problem with economic and societal significance, which goes beyond the particular domain of scallop population assessment, and can possibly extend to other problems of environmental monitoring, or even defense (e.g. mine detection). Given the general unavailability of similar automation tools, the proposed one can have potential impact in the area of underwater automation. The multi-layered approach not only introduces several technical innovations at the implementation level, but also provides a specialized package for benthic habitat assessment. At a processing level, it provides the flexibility to re-task individual data processing layers for different detection applications. When viewed as a complete package, the approach offers an efficient tool to benthic habitat specialists for processing large image datasets.

In the this chapter, we discuss the details of the multi-layered scallop recognition system [7, 8, 9]. This chapter also lists information about the data collection effort that provided the scallop data for the scallop enumeration survey. Finally, an in depth comparison of the differences between this multi-layered framework and an earlier scallop recognition work [10] is discussed.

1.2 Background

1.2.1 Underwater Animal Recognition

In natural settings, living organisms often tend to blend into their environments to evade detection via camouflage. Webster's thesis work [11] provides a detailed exposition on the visual camouflage mechanisms adopted by animals to blend into their background. Under such circumstances of camouflage, there are very limited visual cues that can be used to identify animals. Even in the presence of visual cues, the task of identifying animals from natural scenes is shown to be a cognitively challenging and complex task [12].

Previous efforts to detect animals like plankton [13, 14], clam [15] and a range of other benthic megafauna [16] exist. Most of these methods here are specialized to a specific species, or only tested in controlled environments. In some cases, the methods require specialized apparatus (like in the plankton recognition studies [13, 14]). A series of automated tools like specialized color correction, segmentation and classification modules along with some level of manual expert support, can be combined identification of several marine organisms like sea anemones and sponges from natural image datasets [16].

The existing techniques for marine animal recognition can be broadly divided into methods devised for identifying mobile organisms and methods for sedentary organisms. The former category is useful in dealing with a wide range of sea organisms like the varied species of fish that swim through water. The latter category is less studied. It includes identifying sedentary marine animals like scallops, corrals and sponges. Both categories present their own set of challenges. In the rest of this section we visit the techniques relevant to moving animals and show how they are different from the methods employed for sedentary animals. An overview of the existing literature on recognizing sedentary animals follows, with special emphasis on methods developed for identifying scallops.

1.2.1.1 Methods for Recognition of Moving Underwater Organisms

Recognizing and counting mobile marine life like fish [17, 18, 19] and studies in aquaculture [20] have been attempted. The recurring theme in these efforts involves the use of stationary cameras to detect the presence of moving species, provided that the background can be described by a prior model. This technique of assuming a known background, and using changes in the background as an evidence for the presence of a moving object entering the field of view of a sensor, is called background subtraction. In the marine species identification case, any changes to the background are assumed to be caused by a moving marine organism. The pixels in the image that deviate from the background model can be labeled as the pixels belonging to the organism.

Once a marine organism is detected through background subtraction, then other computer vision or machine learning techniques can be used to classify the organism into a specific species based on its visible characteristics. This classification task can be achieved through conventional machine learning approaches. For instance the salmon species classification algorithm developed by Williams et al. [19] uses active contours to model the shape of the fish before comparing these contours to known salmon species. However, if the pixels corresponding to the organism are contaminated by high levels of noise, a specialized technique that is robust to noise might be required.

Background subtraction requires a mathematical model that describes the distribution of background pixels. In an underwater setting, such a background model can only be obtained if the camera is stationary and is observing a static background, or in the case that the background model represents , the evolution of which over time can be captured through a mathematical model. Such well defined background models are not always available. An opportunity to employ the background subtraction-based techniques arises in underwater environments with stationary fixtures designed to study a specific underwater location. In instances where such stationary arrangement of cameras is not available, background subtraction is inapplicable due to the lack of a background model.

1.2.1.2 Methods for Recognition of Sedentary Underwater Organisms

Since sedentary marine animals like scallops do not typically move (unless chased by a predator), a mobile robotic platform is required to traverse subsea relief to image and recognize those marine animals. Extending background subtraction to work with mobile robotic platforms is challenging, since the motion of the platform causes changes in its background. Generating a model for the background to perform background subtraction in these cases is problematic. This makes the task of detecting sedentary organisms with moving sensors even more challenging than detecting moving organism with stationary sensors. The lack of background model in these cases motivates the development of a foreground model. If a foreground model is available, the task of detecting an organism can be realized as a search for pixels satisfying the foreground model in the image.

Detecting an organism typically involves segmenting all pixels of the organism, in order for one to classify the organism into a known category. The motion-based segmentation of marine animals that involves subtracting a known model of background from a snapshot of the environment, followed by attributing the pixels with non-zero values to the foreground is inapplicable in cases where the background model does not exist. Furthermore, the task of segmentation can be challenging in noisy images with weak edges, since the boundary pixels of the foreground object cannot be easily distinguished from background pixels.

Thus, the lack of background model makes background subtraction problematic. This leads to the need for techniques that depend on foreground models, and use of other features to detect and segment organisms from the background. This task becomes even more complicated if the organism does not present significant visual cues that make it distinctive from the background, as in the case of creatures exhibiting camouflage. High levels of noise or unpredictable environmental variables could also significantly affect the effectiveness of any animal recognition mechanism.

1.2.1.3 Scallop Recognition Methods

There are several aspects that make scallop recognition challenging. Scallops, especially when viewed in low resolution, do not provide features that would clearly distinguish them from their natural environment. This presents a major challenge in designing an automated identification process based on visual data. To compound this problem, visual data collected from the species' natural habitat contain a significant amount of speckle noise. Some scallops are also partially or almost completely covered by sediment, obscuring the scallop shell features. A highly robust detection mechanism is required to overcome these impediments.

There is a range of previously developed methods specialized for scallop recognition [10, 21, 22, 23, 24, 7, 8, 9] that operate on different assumptions, either with regards to the environmental conditions or the quality of data. Existing approaches to automated scallop counting in artificial environments [22, 23] employ a detection mechanism based on intricate distinguishing features like fluted patterns in scallop shells and exposed shell rim of scallops, respectively. Imaging these intricate scallop shell features might be possible in artificial scallop beds with stationary cameras and minimal sensor noise, but this level of detail is difficult to obtain from low resolution images of scallops in their natural environment. A major factor that contributes to the poor image resolution is the fact that sometimes the image of a target is captured several meters away from it. Overcoming this problem by operating an underwater vehicle much closer to the ocean floor will adversely impact the image footprint (i.e. area covered by an image) and increase the risk of damaging the vehicle.

Furthermore, existing work on scallop detection [10, 21] in their natural environment is limited to small datasets (often less than 100 images). A sliding window approach has been used [21] to focus the search for the presence of scallops. The large number of overlapping windows that need to be processed per image raises scalability concerns if this method were to operate on a large dataset containing millions of images. Additionally, the small number of natural images used as a test set raises questions about the generalizibility of this method and its ability to function under

varied environmental conditions. The work by Dawkins [10] is more detailed in its treatment of the natural environmental conditions spanning the scallop habitat. The images used here are collected using a towed camera system that minimizes noise, a fact which greatly enhances the performance of the machine learning and computer vision algorithms. Despite the elaborate imaging setup designed to minimize noise, the results reported are derived only from a few tens to hundreds of images. It is not clear if those machine learning methods [10] can extend to noisy image data captured by Autonomous Underwater Vehicle (AUV)s. From these studies alone, it is not clear if such methods can be used effectively in cases of large datasets comprising several thousand seabed images. An interesting example of machine-learning methods applied to the problem of scallop detection [24] utilizes the concept of Bottom-Up Visual Attention (BUVA). The approach is promising but it does not use any ground truth for validation.

There is more work [7, 8, 9] that offers a multi-layered object recognition framework validated on a natural image dataset for scallop recognition application. The main emphasis there (and in this dissertation) is to develop a technique that can work on low quality noisy sensor data collected using AUVs. The other objective is to build a scalable architecture that can operate on large image datasets in the order of thousands to millions of images and can be generalized for recognizing other marine organisms. A detailed comparison between the scallop recognition approaches in Dawkins et al. [10] and Kannappan et al.[9] is provided in Section 1.8.

1.2.2 Motivation for a Generalized Automated Object Recognition Tool

Understanding the parameters that affect the habitat of underwater organisms is of interest to marine biologists and government officials charged with regulating a multi-million dollar fishing industry. Dedicated marine surveys are needed to obtain population assessments. One traditional scallop survey method, still in use today, is a dredge-based survey. Dredge-based surveys have been extensively used for scallop population density assessment [25]. The process involves dredging part of the ocean

floor, and manually counting the animals of interest found in the collected material. In addition to being invasive and detrimental to the creatures habitat [26], these methods have accuracy limitations and can only generalize population numbers up to a certain extent. There is a need for non-invasive and accurate survey alternatives.

The availability of a range of robotic systems in form of towed camera and Autonomous Underwater Vehicle (auv) systems offer possibilities for such non-invasive alternatives. Optical imaging surveys using underwater robotic platforms provide higher data densities. The large volume of image data (in the order of thousands to millions of images) can be both a blessing and a curse. On one hand, it provides a detailed picture of the species habitat; on the other requires extensive manpower and time to process the data. While improvements in robotic platform and image acquisition systems have enhanced our capabilities to observe and monitor the habitat of a species, we still lack the required arsenal of data processing tools. This need motivates the development of automated tools to analyze benthic imagery data containing scallops.

One of the earliest video based surveys of scallops [27] reports that it took from 4 to 10 hours of tedious manual analysis in order to review and process one hour of collected seabed imagery. The report suggests that an automated computer technique for processing of the benthic images would be a great leap forward; to this time, however, no such system is available. There is anecdotal evidence of in-house development efforts by the HabCam group [28] towards an automated system but as yet no such system has emerged to the community of researchers and managers. A recent manual count of our AUV-based imagery dataset indicated that it took an hour to process 2080 images, whereas expanding the analysis to include all benthic macro-organisms reduced the rate down to 600 images/hr [29]. Another manual counting effort [30] reports a processing time of 1 to 10 hours per person to process each image tow transect (the exact image number per tow was not reported). The same report indicates that the processing time was reduced to 12 hours per tow by subsampling 1% of the images.

Future benthic studies can be geared towards increasing data densities with

the help of robotic optical surveys. It is clear that the large datasets, in the order of millions of images, generated by these surveys will impose a strain on researchers if the images are to be process manually. This strongly suggests the need for automated tools that can process underwater image datasets. Motivated by the need to reduce human effort, Schoening [16] has proposed a range of tools that can be generalized to organisms like sea-anemones. With an additional requirement of being able to work with low-resolution noisy underwater images, a generalized multi-layered framework that can be used to detect and count underwater organisms has been proposed [7, 8, 9]. This method has been evaluated on a scallop population assessment effort on a dataset containing over 8000 images, the details of which is the subject of this chapter.

1.3 Preliminaries

1.3.1 Visual Attention

Visual attention is a neuro-physiologically inspired machine learning method [31] that attempts to mimic the human brain function in its ability to rapidly single out objects that are different from their surroundings within imagery data. The method is based on the hypothesis that the human visual system first isolates points of interest in an image, and then sequentially processes these points based on the degree of interest associated with each point. The degree of interest associated with a pixel is called *salience*, and points with the highest salience values are processed first. The method is used to pinpoint regions in an image where the value of some pixel attributes may be an indicator to its uniqueness relative to the rest of the image.

According to the visual attention hypothesis [31], in the human visual system the input video feed is split into several feature streams. Locations in these feature streams that are different from others in their neighborhood would generate peaks in the *center-surround* feature maps. The different center-surround feature maps can be combined to obtain a saliency *map*. Peaks in these resulting saliency maps, otherwise known as *fixations*, become points of interest, processed sequentially in descending order of their salience values.

Itti et al. [32] proposed a computational model for visual attention. According to this model, an image is first processed along three feature streams (color, intensity, and orientation). The color stream is further divided into two sub-streams (red-green and blue-yellow) and the orientation stream into four sub-streams ($\theta \in \{0°, 45°, 90°, 135°\}$). The image information in each sub-stream is further processes in 9 different scales. In each scale, the image is scaled down using a factor $\frac{1}{2^k}$ (where $k = 0, \ldots, 8$), resulting in some loss of information as scale increases. The resulting image data for each scale factor constitutes the *spatial scale* for the particular sub-stream.

The sub-stream feature maps are compared across different scales to expose differences in them. Through the spatial scales in each sub-stream feature map, the scaling factors change the information contained. Resizing these spatial scales to a common scale through interpolation, and then comparing them, brings out the mismatch between the scales. Let $\ominus$ be an pixel operator that takes pixel-wise differences between resized sub-streams. This function is called the *center-surround* operator, and codifies the mismatches in the differently scaled sub-streams in the form of another map: the center-surround feature map. In the case of the intensity stream, with $c \in \{2, 3, 4\}$ and $s = c + \delta$ for $\delta \in \{3, 4\}$ denoting the indices of two different spatial scales, the center-surround feature map is given by

$$I(c, s) = |I(c) \ominus I(s)| \quad . \tag{1.1}$$

Similarly center-surround feature maps are computed for each sub-stream in color and orientation streams.

In this way, the seven sub-streams (two in color, one in intensity and four in orientation), yield a total of 42 center-surround feature maps. All center-surround feature maps in an original stream (color, intensity, and orientation) are then combined into a *conspicuity map* (CM): one for color $\bar{C}$, one for intensity $\bar{I}$, and one for orientation $\bar{O}$. Define the cross-scale operator $\oplus$ that adds up pixel values in different maps. Let w_{cs} be scalar weights associated with how much the combination of two different spatial scales c and s contributes to the resulting conspicuity map. If M is the global maximum

over the map resulting from the $\oplus$ operation, and $\bar{m}$ is the mean over all local maxima present in the map, let $\mathcal{N}(\cdot)$ be a normalization operator that scales that map by a factor of $(M - \bar{m})^2$. For the case of intensity, this combined operation produces a conspicuity map based on the formula

$$\bar{I} = \bigoplus_{c=2}^{4} \bigoplus_{s=c+3}^{c+4} w_{cs}\, \mathcal{N}(I(c,s)) \ . \tag{1.2}$$

The three conspicuity maps—for intensity, color and orientation—are combined to produce the *saliency map*. If scalar weights for each data stream are selected, say $w_{\bar{I}}$ for intensity, $w_{\bar{C}}$ for color, and $w_{\bar{O}}$ for orientation, the saliency map can be expressed mathematically as

$$S = w_{\bar{I}}\, \mathcal{N}(\bar{I}) + w_{\bar{C}}\, \mathcal{N}(\bar{C}) + w_{\bar{O}}\, \mathcal{N}(\bar{O}) \ . \tag{1.3}$$

In a methodological variant of visual attention known as BUVA, all streams are weighted equally: w_{cs} is constant for all $c \in \{2,3,4\}$, $s = c + \delta$ ($\delta \in \{3,4\}$) and $w_{\bar{I}} = w_{\bar{C}} = w_{\bar{O}}$. A winner-takes-all neural network is typically used [32, 33] to compute the maxima, or fixations, on this map—other discrete optimization methods are of course possible. In the context of visual attention, fixations are the local maxima of the saliency map. These fixations lead to shifts in *focus of attention*, or in other words, enables the human vision processing system to preferentially process regions around fixations in an image.

In a different variant of visual attention referred to as TDVA [34], the weights in (1.2) and (1.3) are selected judiciously to bias fixations toward particular attributes. There exists a method to select these weights in the general case when N_m maps are to be combined with those weights [34]. Let N be the number of images in the learning set, and N_{iT} and N_{iD} be the number of targets—in this case, scallops—and distractors (similar objects) in image i within the learning set. For image i, let P_{ijT_k} denote the local maximum of the numerical values of the map for feature j in the neighborhood of the target indexed k; similarly, let P_{ijD_r} be the local maximum of the numerical values

Figure 1.1: [8] Seabed image with scallops shown in red circles

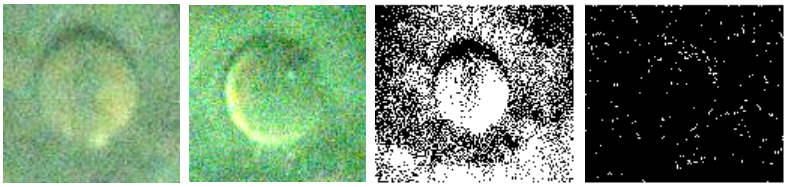
of the map for feature j in the neighborhood of distractor indexed r. The weights for a combination of maps are determined by

$$w'_j = \frac{\sum_{i=1}^{N} N_{iT}^{-1} \sum_{k=1}^{N_{iT}} P_{ijT_k}}{\sum_{i=1}^{N} N_{iD}^{-1} \sum_{r=1}^{N_{iD}} P_{ijD_r}}$$
$$w_j = \frac{w'_j}{\frac{1}{N_m} \sum_{j=1}^{N_m} w'_j}, \tag{1.4}$$

where $j \in \{1, \ldots, N_m\}$ is the index set of the different maps to be combined. Equations (1.4) are used for the selection of weights w_{cs} in (1.2), and $w_{\bar{I}}$, $w_{\bar{O}}$, $w_{\bar{C}}$ in (1.3).

1.4 Problem Statement

A visual scallop population assessment process involves identifying these animals in image datasets. A representative example of an image from the dataset we had to work with is shown in Figure 1.1 (scallops marked within red circles). A general solution to automated image annotation might not necessarily be effective for the dataset at hand. The need here is to identify algorithms and methods that will work best under *poor* lighting and imaging conditions, characteristic of this particular scallop counting application. The results from using elementary image processing methods like thresholding and edge detection on the images (see Figure 1.2c and 1.2d) demonstrate



(a) (b) (c) (d)

Figure 1.2: [9] (a) Scallop with yellowish tinge and dark crescent; (b) Scallop with yellowish tinge and bright shell rim crescent; (c) Scallop with no prominent crescents and texturally identical to the background (d) Scallop sample after thresholding; (c) Scallop sample after edge detection.

the need for a more sophisticated approach (possibly a hybrid combination of several techniques).

Another challenge, related to the issue of low image resolution and high levels of speckle noise, is the selection of appropriate scallop features that would enable distinguishing between these organisms and other objects. In the particular dataset, one recurrent visual pattern is a dark crescent on the upper perimeter of the scallop shell, which is the shadow cast by the upper open scallop shell produced from the AUV strobe light (see Figure 1.2a). Another pattern that could serve as a feature in this dataset is a bright crescent on the periphery of the scallop, generally associated with the visible interior of the bottom half when the scallop shell is partly open (see Figure 1.2b). A third pattern may be a yellowish tinge associated with the composition of the scallop image (see Figure 1.2b).

We have leveraged visual patterns [8] to develop a three-layered scallop counting framework that combines tools from computer vision and machine learning. This particular hybrid architecture uses top-down visual attention, graph-cut segmentation and template matching along with a range of other filtering and image processing techniques. Though this architecture offers a performance of over 63% true positive detection rate, it has a very large number of false positives. To mitigate this problem, we extend the framework [8] by adding a fourth, false-positives filtering layer [9].

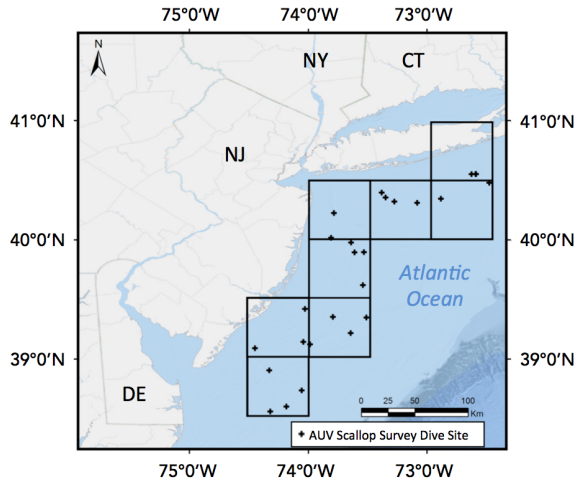


Figure 1.3: Map of the survey region from Shinnecock, New York to Cape May, New Jersey, divided into eight blocks or strata

1.5 Scallop Survey Procedure

The 2011 RSA project (Titled: "A Demonstration Sea Scallop Survey of the Federal Inshore Areas of the New York Bight using a Camera Mounted Autonomous Underwater Vehicle") was a proof-of-concept project that successfully used a digital, rapid-fire camera integrated to a Gavia AUV, to collect a continuous record of photographs for mosaicking, and subsequent scallop enumeration. In July 2011, transects were completed in the northwestern waters of the mid-Atlantic Bight at depths of 25-50 m. The AUV continuously photographed the seafloor along each transect at a constant distance of 2 m above the seafloor. Parallel sets of transects were spaced as close as 4 m. Georeferenced images were manually analyzed for the presence of sea scallops using position data logged (using Doppler Velocity Log (DVL) and Inertial Navigation System (INS)) with each image.

1.5.1 Field Survey Process

In the 2011 demonstration survey, the federal inshore scallop grounds from Shinnecock, New York to Ocean View, Delaware, was divided into eight blocks or strata (as shown in Figure 1.3). The *f/v Christian and Alexa* served as the surface support platform from which a Gavia AUV (see Figure 1.4) was deployed and recovered. The

AUV conducted photographic surveys of the seabed for a continuous duration of approximately 3 hours during each dive, repeated 3–4 times in each stratum, with each stratum involving roughly 10 hours of imaging and an area of about 45 000 m^2. The AUV collected altitude (height above the seabed) and attitude (heading, pitch, roll) data, allowing the georectification of each image into scaled images for size and counting measurements. During the 2011 pilot study survey season, over 250 000 images of the seabed were collected. These images were analyzed in the University of Delaware's Coastal Sediments, Hydrodynamics and Engineering Laboratory for estimates of scallop abundance and size distribution. The *f/v Christian and Alexa* provided surface support, and made tows along the AUV transect to ground-truth the presence of scallops and provide calibration for the size distribution. Abundance and sizing estimates were computed manually for each image using a GUI-based digital sizing software. Each image included embedded metadata that allowed it to be incorporated into existing benthic image classification systems (HabCam mip [10]).

During this proof of concept study, in each stratum the f/v *Christian and Alexa* made one 15-minute dredge tow along the AUV transect to ground-truth the presence of scallops and other fauna, and provide calibration for the size distribution. The vessel was maintained on the dredge track by using Differential GPS. The tows were made with the starboard 15 ft (4.572 m) wide New Bedford style commercial dredge at the commercial dredge speed of 4.5–5.0 knots. The dredge was equipped with 4 inch (10.16 m) interlocking rings, an 11 inch (27.94 cm) twine mesh top, and turtle chains. After dredging, the catch was sorted, identified, and weighed. Length-frequency data were obtained for the caught scallops. This information was recorded onto data logs and then entered into a laptop computer database aboard ship for comparison to the camera image estimates.

The mobile platform of the AUV provided a more expansive and continuous coverage of the seabed compared to traditional fixed drop camera systems or towed camera systems. In a given day, the AUV surveys covered about 60 000 m^2 of seabed from an altitude of 2 m above the bed, simultaneously producing broad sonar swath

coverage and measuring the salinity, temperature, dissolved oxygen, and chlorophyll-a in the water.

1.5.2 Sensors and Hardware

The University of Delaware AUV (Figure 1.4) was used to collect continuous images of the benthos, and simultaneously map the texture and topography of the seabed. Sensor systems associated with this vehicle include: (1): a 500 kHz GeoAcoustics GeoSwath Plus phase measuring bathymetric sonar; (2): a 900/1800 kHz Marine Sonic dual-frequency high-resolution side-scan sonar; (3): a Teledyne Rd Instruments 1200 kHz acoustic doppler velocity log (DVL)/Acoustic doppler current profiler (ADCP); (4): a Kearfott T-24 inertial navigation system; (5): an Ecopuck flntu combination fluorometer / turbidity sensor; (6): a Point Grey Scorpion model 20SO digital camera and LED strobe array; (7): an Aanderaa Optode dissolved oxygen sensor; (8): a temperature and density sensor; and, (9): an altimeter. Each sensor separately records time and spatially stamped data with frequency and spacing. The AUV is capable of very precise dynamic positioning, adjusting to the variable topography of the seabed while maintaining a constant commanded altitude offset.

1.5.3 Data Collection

The data was collected over two separate five-day cruises in July 2011. In total, 27 missions were run using the AUV to photograph the seafloor (For list of missions see Table 1.1). Mission lengths were constrained by the 2.5 to 3.5 hour battery life of the AUV. During each mission, the AUV was instructed to follow a constant height of 2 m above the seafloor. In addition to the 250 000 images that were collected, the AUV also gathered data about water temperature, salinity, dissolved oxygen, geoswath bathymetry, and side-scan sonar of the seafloor.

The camera on the AUV, a Point Grey Scorpion model 20SO (for camera specifications see Table 1.2), was mounted inside the nose module of the vehicle. It was focused at 2 m, and captured images at a resolution of 800×600. The camera lens

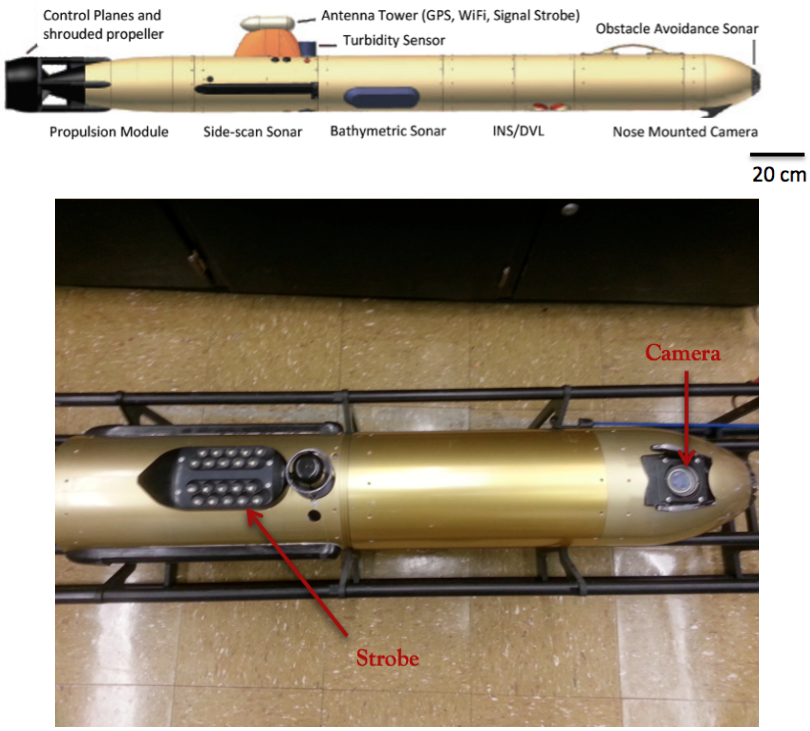


Figure 1.4: Schematics and image of the Gavia AUV

had a horizontal viewing angle of 44.65 degrees. Given the viewing angle and distance from the seafloor, the image footprint can be calculated as 1.86×1.40 m^2. Each image was saved in jpeg format, with metadata that included position information (including latitude, longitude, depth, altitude, pitch, heading and roll) and the near-seafloor environmental conditions analyzed in this study. This information is stored in the header file, making the images readily comparable and able to be incorporated into existing RSA image databases, such as the HabCam database. A manual count of the number of scallops in each image was performed and used to obtain overall scallop abundance assessment. Scallops counted were articulated shells in life position (left valve up) [29].

1.6 Methodology

The multi-layered scallop counting framework that comprises four layers of processing on underwater images for the purpose of obtaining scallop counts is discussed

Mission	Number of images
LI1[1]	12 775
LI2	2 387
LI3	8 065
LI4	9 992
LI5	8 338
LI6	11 329
LI7	10 163
LI8	9 780
LI9	2 686
NYB1[2]	9 141
NYB2	9 523
NYB3	9 544
NYB4	9 074
NYB5	9 425
NYB6	9 281
NYB7	12 068
NYB8	9 527
NYB9	10 950
NYB10	9 170
NYB11	10 391
NYB12	7 345
NYB13	6 285
NYB14	9 437
NYB15	11 097
ET1[3]	9 255
ET2	12 035
ET3	10 474

[1] LI–Long Island
[2] NYB–New York Bight
[3] ET–Elephant Trunk

Table 1.1: List of missions and number of images collected

Attribute	Specs
Name	Point Grey Scorpion 20SO Low Light Research Camera
Image Sensor	8.923 mm Sony ccd
Horizontal Viewing Angle	44.65 degrees (underwater)
Mass	125 g
Frame rate	3.75 fps
Memory	Computer housed in AUV nose cone
Image Resolution	800×600
Georeferenced metadata	Latitude, longitude, altitude, depth
Image Format	jpeg

Table 1.2: Camera specifications

in this section. The four layers involve the sequential application of Top-Down Visual Attention, Segmentation, Classification and False-Positive Filtering.

1.6.1 Layer I: Top-Down Visual Attention

1.6.1.1 Learning

A customized TDVA algorithm can be designed to sift automatically through the body of imagery data, and focus on regions of interest that are more likely to contain scallops. The process of designing the TDVA algorithm is described below.

The first step is a small-scale, BUVA based saliency computation. The saliency computation is performed on a collection of randomly selected 243 annotated images, collectively containing 300 scallops. This collection constitutes the *learning set*. Figure 1.5 represents graphically the flow of computation and shows the type of information in a typical image that visual attention tends to highlight.

A process of extremum seeking on the saliency map of each image identifies fixations in the associated image. If a 100×100 pixel window—corresponding to an approximately 23×23 cm^2 area on the seafloor—centered around a fixation point contained the center of a scallop, the corresponding fixation was labeled a *target*; otherwise, it is considered a *distractor*.

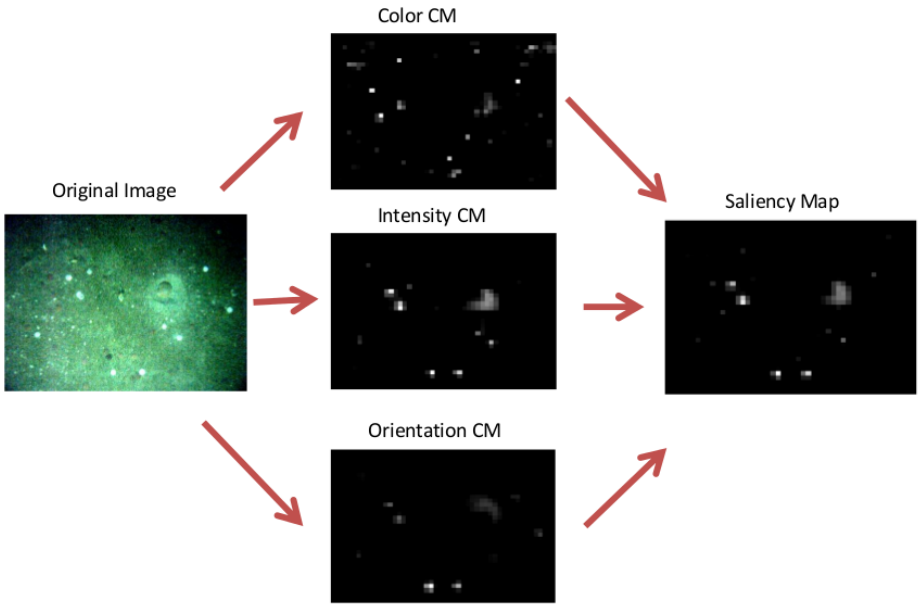


Figure 1.5: Illustration of computation flow for the construction of saliency maps

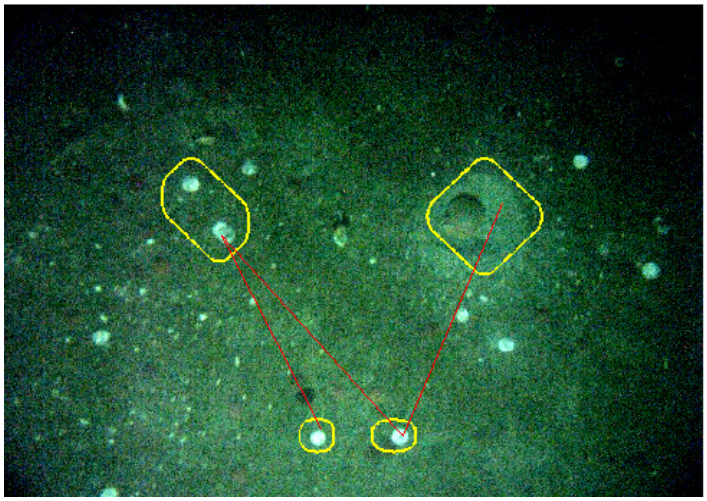


Figure 1.6: Illustration of fixations (marked by yellow boundaries): red lines indicate the order in which the fixations were detected with the lower-left fixation being the first.

Table 1.3: Top-down weights for feature maps

		Center Surround Feature Scales					
		1	2	3	4	5	6
Color	red-green	0.8191	0.8031	0.9184	0.8213	0.8696	0.7076
	blue-yellow	1.1312	1.1369	1.3266	1.2030	1.2833	0.9799
Intensity	intensity	0.7485	0.8009	0.9063	1.0765	1.3111	1.1567
Orientation	0°	0.7408	0.2448	0.2410	0.2788	0.3767	2.6826
	45°	0.7379	0.4046	0.4767	0.3910	0.7125	2.2325
	90°	0.6184	0.5957	0.5406	1.2027	2.0312	2.1879
	135°	0.8041	0.6036	0.7420	1.5624	1.1956	2.3958

The target and distractor regions are determined in all the feature and conspicuity maps for each one of these processed images in the learning set. This is done by adaptively thresholding and locally segmenting the points around the fixations with similar salience values in each map. Then the mean numerical value in neighborhoods around these target and distractor regions in the feature maps and conspicuity maps are computed. These values are used to populate the P_{ijT_k} and P_{ijD_r} variables in (1.4), and determine the top-down weights for feature maps and conspicuity maps.

For the conspicuity maps, the center-surround scale weights w_{cs} computed through (1.4) and consequently used in (1.2), are shown in Table 1.3. For the saliency map computation, the weights resulting from the application of (1.4) on the conspicuity maps are $w_{\bar{I}} = 1.1644$, $w_{\bar{C}} = 1.4354$ and $w_{\bar{O}} = 0.4001$. The symmetry of the scallop shell in our low-resolution dataset justifies the relatively small value of the orientation weight.

1.6.1.2 Implementation and Testing

To test the performance of the customized TDVA algorithm, it is applied on two image datasets, the size of which is shown in Table 1.5. In this application, the saliency maps are computed via the formulae (1.3) and (1.2), using the weights listed in Table 1.3. Convergence time of the winner-takes-all neural network that finds fixations in the

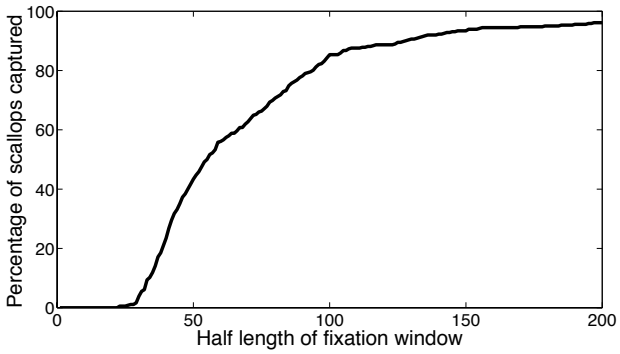
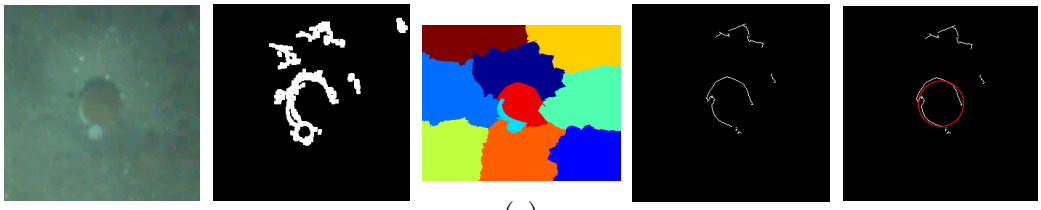


Figure 1.7: Percentage of scallops enclosed in the fixation window as a function of window half length (in pixels)



(a) (b) (c) (d) (e)

Figure 1.8: a Fixation window from layer I; b Edge segmented image; c graph-cut segmented image; d Region boundaries obtained when the edge segmented image is used as a mask over the graph-cut segmented image boundaries; e circle fitted on the extracted region boundaries.

saliency map of each image in the datasets of Table 1.5, is controlled using dynamic thresholding: It is highly unlikely that a fixation that contains an object of interest requires more than 10 000 iterations. If convergence to some fixation takes more than this number of iterations, then the search is terminated and no more fixations are sought in the image.

Given that an image in datasets of Table 1.5 contains two scallops on average, no more than ten fixations are sought in each image (The percentage of images in the datasets that contained more than 10 scallops was 0.002%). Since in the testing phase the whole scallop—not just the center—needs to be included in the fixation window, the size of this window is set at 270×270 pixels; more than 91% of the scallops are accommodated inside the window (Figure 1.7).

1.6.2 Layer II: Segmentation and shape extraction

This processing layer consists of three separate sub-layers: edge based segmentation (involves basic morphological operations like smoothing, adaptive thresholding and edge detection), graph-cut segmentation, and shape fitting. The flow of the segmentation process for a typical fixation window containing scallop is illustrated in Figure 1.8. Figure 1.8a shows a fixation window. Edge-based segmentation on this window yields the edge segmented image of Figure 1.8b. At the same time, graph-cut segmentation process [35] is applied on the fixation window to decompose it into 10 separate regions as seen in Figure 1.8c. The boundaries of these segments are matched with the edges in the edge segmented image. This leads to further filtering of the edges, and eventually to the region boundaries on Figure 1.8d. This is followed by fitting a circle to each of the contours in the filtered region boundaries (Figure 1.8d). Only circles with dimensions close to that of a scallop (diameter $20-70$ pixels) are retained (Figure 1.8e), which in turn helps in rejection of other non-scallop round objects.

The choice of the shape to be fitted is suggested by the geometry of the scallop's shell. Finding the circle that fits best to a given set of points is formulated as an optimization problem [36, 37].

Given a set of n points on a connected contour each with coordinates (x_i, y_i) $(i \in \{1, 2, \ldots, n\})$, define a function of four parameters A, B, C, and D:

$$F_2(A, B, C, D) = \frac{\sum_{i=1}^{n}[A(x_i^2 + y_i^2) + Bx_i + Cy_i + D]^2}{n^{-1}\sum_{i=1}^{n}[4A^2(x_i^2 + y_i^2) + 4ABx_i + 4ACy_i + B^2 + C^2]} \ . \tag{1.5}$$

It is shown [36] that minimizing (1.5) over these parameters yields the circle that fits best around the contour. The center (a, b) and the radius of this best-fit circle are given as a function of the parameters as follows:

$$a = -\frac{B}{2A} \ , \qquad b = -\frac{C}{2A} \ , \qquad R = \sqrt{\frac{B^2 + C^2 - 4AD}{4A^2}} \ . \tag{1.6}$$

For all annotated scallops in the testing image dataset, the quality of the fit is quantified by means of two scalar measures: the center error e_c, and the percent radius error e_r. An annotated scallop would be associated with a triple (a_g, b_g, R_g)—the coordinates of its center (a_g, b_g) and its radius R_g. Using the parameters of the

fit in (1.6), the error measures are evaluated as follows, and are required to be below the thresholds specified on the right hand side in order for the scallop to be considered detected.

$$e_c = \sqrt{(a_g - a)^2 + (b_g - b)^2} \leq 12 \ \text{(pixels)} \qquad e_r = \frac{|R_g - R|}{R_g} \leq 0.3 \ .$$

These thresholds were set empirically, taking into account that radius measurements in manual counts used as ground truth [29] have a measurement error of 5–10%.

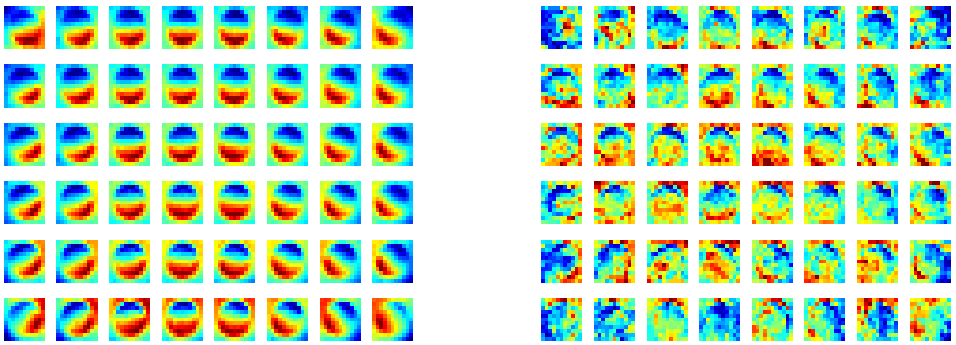
1.6.3 Layer III: Classification

The binary classification problem solved in this layer consists of identifying specific features in the images which mark the presence of scallops. These images are obtained by a using a camera at the nose of the AUV, illuminated by a strobe light close to its tail (mounted to the hull of the control module at an oblique angle to the camera). Our hypothesis is that due to this camera-light configuration, scallops appear in the images with a bright crescent at the lower part of its perimeter and a dark crescent at the top—a shadow. Though crescents appear in images of most scallops, their prominence and relative position with respect to the scallop varies considerably. The hypothesis regarding the origin of the light artifacts implies that the approximate profile and orientation of the crescents is a function of their location in the image.

1.6.3.1 Scallop Profile Hypothesis

A statistical analysis was performed on a dataset of 3 706 manually labeled scallops (each scallop is represented as (a, b, R) where a, b are the horizontal and vertical coordinates of the scallop center, and R is its radius). For this analysis, square windows of length $2.8 \times R$ centered on (a, b) were used to crop out regions from the images containing scallops.[2] Each cropped region was filtered in grayscale, contrast stretched,

[2] Using a slightly larger window size ($> 2 \times R$, the size of the scallop) includes a neighborhood of pixels just outside the scallop which is where crescents are expected. This also improves the performance of local contrast enhancement, leading to better edge detection.



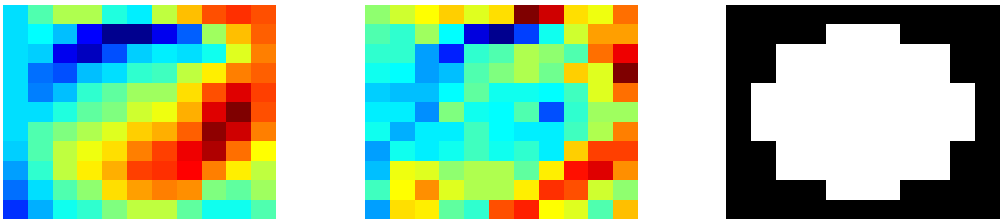
(a) (b)

Figure 1.9: a Mean map of scallops in each quadrant b Standard deviation map of scallops in each quadrant. Red corresponds to higher numeric values and blue correspond to lower numeric values.

and then normalized by resizing to 11×11 dimension or 121 bins. To show the positional dependence of the scallop profiles, the image plane is discretized into 48 regions (6×8 grid). Scallops whose centers lie within each grid square are segregated. The mean (Figure 1.9a) and standard deviation (Figure 1.9b) of the 11×11 scallop profiles of all scallops per grid square over the whole dataset of 3 706 images was recorded. The lower standard deviation found in the intensity maps of the crescents on the side of the scallop facing away from the camera reveal that these artifacts are more consistent as markers compared to the ones closer to the lens.

1.6.3.2 Scallop Profile Learning

The statistics of the dataset of 3 706 images used to produce Figure 1.9 form a look-up table that represents reference scallop profile (mean and standard deviation maps) as a function of scallop center pixel location. To obtain the reference profile for a pixel location, the statistics from all the scallops whose centers lie inside a 40×40 window centered on the pixel is used. This look-up table can be compressed; it turns out that not all of the 121 bins (11×11) within each map is equally informative, because bins close to the boundary are more likely to include a significant number of background pixels. For this reason, a circular mask with a radius covering 4 bins is



(a) (b) (c)

Figure 1.10: Intensity statistics and mask for a region centered at a pixel with coordinates $(470, 63)$ in the image a Map of mean intensity; b Map of intensity standard deviation; c Mask applied to remove background points.

applied to each map (Figure 1.10), thus reducing the number of bins that are candidates as features for identification to 61. Out of these 61 bins, 15 additional bins having the highest standard deviation are ignored, leading to a final set of 46 bins. The value in the selected 46 bins from mean map forms a 46-dimensional feature vector associated with that region. The corresponding 46 bins from the standard deviation map are also recorded, and are used to weight the features (as seen later in (1.7)).

1.6.3.3 Scallop Template Matching

With this look-up table that codes the reference scallop profile for every scallop center pixel location, the resemblance of any segmented object to a scallop can now be assessed. The metric used for this comparison is a weighted distance function between the elements of the feature vector for the region corresponding to the segmented object, and that coming from the look-up table, depending on the location of the object in the image being processed. If this distance metric is below a certain threshold D_{thresh}, the object is classified a scallop. Technically, let $X^o = (X^o_1, X^o_2, \ldots, X^o_{46})$ denote the feature vector computed for the segmented object, and $X^s = (X^s_1, \ldots, X^s_{46})$ the reference feature vector. Every component of the X^s vector is a reference mean intensity value for a particular bin, and is associated with a standard deviation σ_k from the reference standard deviation map. To compute the distance metric, first normalize X^o to produce

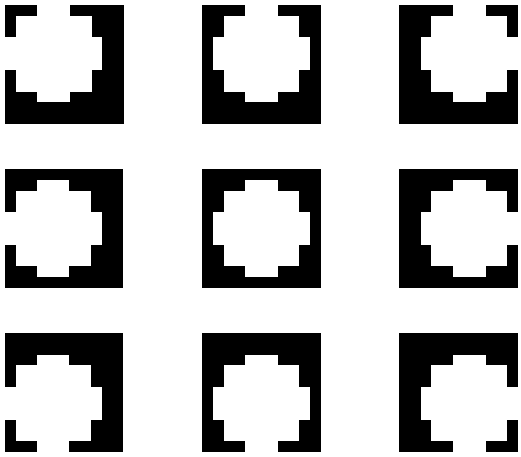


Figure 1.11: Nine different masks slightly offset from the center used to make the classification layer robust to errors in segmentation

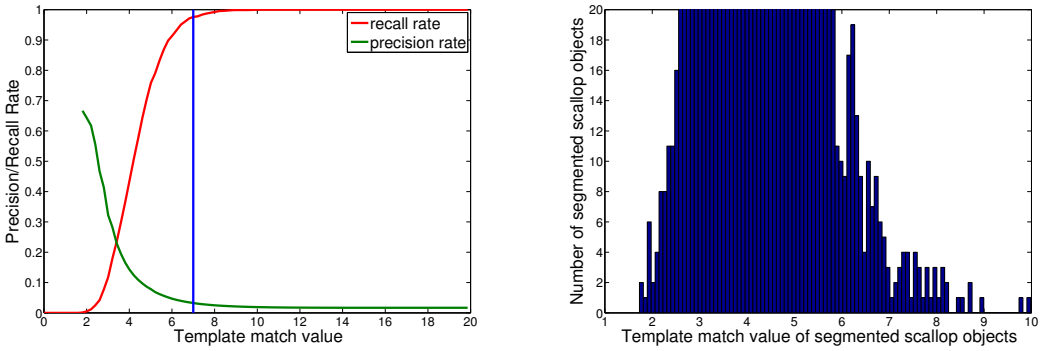
vector $X^{\bar{o}}$ with components

$$X_p^{\bar{o}} = \min_k X_k^s + \left(\frac{\max_k X_k^s - \min_k X_k^s}{\max_k X_k^o - \min_k X_k^o} \right) \left[X_p^o - \min_k X_k^o \right] \text{ for } p = 1, \ldots, 46 \ ,$$

and then evaluate the distance metric D_t quantifying the dissimilarity between the normalized object vector $X^{\bar{o}}$ and the reference feature vector X^s as

$$D_t = \sqrt{\sum_{k=1}^{n} \frac{\|X_k^{\bar{o}} - X_k^s\|^2}{\sigma_k}} \ . \tag{1.7}$$

Small variations in segmentation can produce notable deviations in the computed distance metric (1.7). To alleviate this effect, the mask of Figure 1.10c was slightly shifted in different directions and the best match in terms of the distance was identified. This process enhanced the robustness of the classification layer with respect to small segmentation errors. Specifically, nine slightly shifted masks were used (shown in Figure 1.11). Out of the nine resulting distance metrics $D_t^{o_1} \ \ldots D_t^{o_9}$, the smallest $D_{\mathsf{obj}} = \min_{p \in \{1,\ldots,9\}} D_t^{o_p}$ is found and used for classification. If $D_{\mathsf{obj}} < D_{\mathsf{thresh}}$, the corresponding object is classified as a scallop. Based on Figures 1.12a–1.12b, the threshold



(a) (b)

Figure 1.12: a Precision-Recall curve with D_{thresh} shown as a vertical line; b Histogram of template match of segmented scallop objects.

value was chosen at $D_{\mathsf{thresh}} = 7$ to give a recall[3] rate of 97%. Evident in Figure 1.12a is the natural trade-off between increasing recall rates and keeping the number of false positives low.

1.6.4 Layer IV: False Positives Filter

To decrease the false positives that are produced in the classification layer, two methods are evaluated as possible candidates: a high-dimensional Weighted Correlation Template Matching (WCTM) technique and a Histogram of Gradients (HOG) method. The main objective here is to find a method that will retain a high percentage of true positive scallop and at the same time eliminate as many false positives from the classification layer as possible.

1.6.4.1 High-dimensional weighted correlation template matching (WCTM)

In this method, the templates used are generated from scallop images that are *not* preprocessed, i.e., images that are not median-filtered, unlike the images that were processed by the first three layers. The intuition behind this is that although median

[3] *Recall* refers to the fraction of relevant instances identified: fraction of scallops detected over all ground truth scallops; *precision* is the fraction of the instances returned that are really relevant compared to all instances returned: fraction of true scallops over all objects identified as scallops.

filtering reduces speckle noise and may improve the performance of segmentation, it also weakens the edges and gradients in an image. Avoiding median filtering helps to generate templates that are more accurate than the ones already used in the classification layer.

Based on the observation that the scallop templates are dependent on their position in the image (Figure 1.9), a new scallop template is generated for each object that is classified as a scallop in Layer III. As indicated before, such an object would be represented by a triplet (a_o, b_o, R_o), where a_o and b_o represent the spatial Cartesian coordinates of object's geometric center, and R_o gives its radius. The representative scallop template is now generated from all scallops in the learning set (containing 3 706 scallops), of which the center is within a 40×40 window in the neighborhood of the object center (a_o, b_o). Each of these scallops is then extracted using a window of size $2.5R \times 2.5R$ where R is the scallop radius. Since these scallops in the learning set can be of different dimensions, it is resized (scaled) to a window of size $2.5R_o \times 2.5R_o$. All these scallop instances in the learning set are finally combined through a pixel-wise mean to obtain the mean representative template. Similarly, a standard deviation map that captures the standard deviation of each pixel in the mean template is also obtained. The templates produced here are of larger size compared to the templates in Layer III (recall that a Layer III template was of size 11×11). The inclusion of slightly more information contributes to these new larger templates being more accurate.

In a fashion similar to the analysis in Layer III, the templates and object pixels first undergo normalization and mean subtraction. Then they are compared. Let $v = (2.5R_o)^2$ be the total number of pixels in both the template and the object, and let the new reference scallop feature (template) and the object be represented by vectors $X^t = (X_1^t, X_2^t, \ldots, X_v^t)$ and $X^u = (X_1^u, \ldots, X_v^u)$, respectively. In addition, let σ be the standard deviation vector associated with X^t. Then the reference scallop feature vector X^t would first be normalized as follows:

$$X_p^{t'} = \min_k X_k^u + \left(\frac{\max_k X_k^u - \min_k X_k^u}{\max_k X_k^t - \min_k X_k^t} \right) \left[X_p^t - \min_k X_k^t \right] ,$$

where p denotes the position of component X_p^t in vector X^t. Normalization is followed by mean subtraction, this time both for the template and for the object. The resulting, mean-subtracted reference scallop feature $X^{\bar{t}}$, and object $X^{\bar{u}}$ are computed as

$$X_p^{\bar{t}} = X_p^{t'} - \frac{1}{v}\sum_{k=1}^{v} X_k^{t'} \ , \qquad\qquad X_p^{\bar{u}} = X_p^{u} - \frac{1}{v}\sum_{k=1}^{v} X_k^{u} \ .$$

Now the standard deviation vector is normalized:

$$\bar{\sigma}_p = \frac{\sigma_p}{\sum_{k=1}^{v}\sigma_k} \ .$$

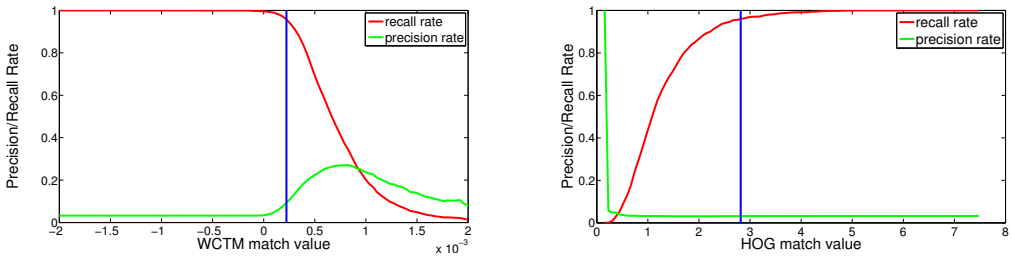
At this point, a metric that expresses the correlation between the mean-subtracted template and the object can be computed. This metric is weighted by the (normalized) variance of each feature. In general, the higher the value of this metric, the better the match between the object and the template. The WCTM similarity metric is given by

$$D_{\mathsf{wctm}} = \sum_{k=1}^{v} \frac{X_k^{\bar{t}} X_k^{\bar{u}}}{\bar{\sigma}_k} \ .$$

The threshold set for the weighted correlation metric D_{wctm}, in order to distinguish between likely true and false positives is at 0.0002222, i.e., any object with a similarity score lower than this threshold is rejected. This threshold value is justified from the precision-recall curves (see Figure 1.13a) of the weighted correlation metric values for the objects filtering down from the classification layer. The threshold shown by the blue line corresponds to 96% recall rate, i.e., 96% of the true positive scallops from the classification layer pass through WCTM. At the same time, WCTM decreases the false positives by over 63%.

1.6.4.2 Histogram of Gradients (HOG)

The HOG feature descriptor encodes an object by capturing a series of local gradients in neighborhood of the object pixels. These gradients are then transformed into a histogram after discretization and normalization. There are several variants of HOG feature descriptors. The R-HOG used for human detection in [38] was tested here as a possible Layer IV candidate.



(a) (b)

Figure 1.13: Precision recall curve for Layer IV candidate methods (a) WCTM and (b) HOG. The blue line marks thresholds $D_{\mathsf{wctm}} = 0.0002222$ and $D_{\mathsf{hog}} = 2.816$. It is important to note that WCTM is a similarity measure and HOG is a dissimilarity measure. This implies that only instances below the indicated threshold D_{wctm} in WCTM, and likewise instances above the threshold D_{hog} in HOG, are rejected as false positives.

To produce R-HOG, the image is first tiled into a series of 8×8 pixel groups referred to here as cells (the image dimensions need to be multiples of 8). The cells are further divided into a series of overlapping blocks each containing groups of 2×2 cells. For each cell a set of 64 gradient vectors (one per pixel) is computed. Each gradient vector contains a direction and magnitude component. In the gradient directions, the sign is ignored reducing the range of angles from 0–360 down to 0–180. The gradient vectors are then binned into a 9–bin histogram ranging from 0–180 degrees with a bin width of 20 degrees. The contribution of each gradient vector is computed as half its gradient magnitude. The other half of the gradient magnitude is split between the two neighboring bins (in case of boundary bins, the neighbors are determined by wrapping around the histogram). The histograms from the 4 cells in each block is then concatenated to get vector v of 36 values (9 per cell). These vectors from each block are then normalized using their L_2-norm; for a vector v this normalization would be expressed as

$$\bar{v} = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}}$$

where ϵ is a small constant (here $\epsilon = 0.01$). The normalized vector $\bar{v}$ from each block is concatenated into a single feature vector F to get the HOG descriptor for the input image.

Since this method imposes a constraint on the image dimensions being multiples of 8, the learning samples (each cropped using a square window of size of $3 \times$ radius) are resized to 24×24. Here, we have to use both positive and negative object samples, the latter being objects other than scallops picked up in the segmentation layer. A HOG feature vector F of length 144 (4 blocks $\times$ 4 cells $\times$ 9 values) is computed for each object instance obtained from the classification layer.

Now several different machine learning methods can be applied, using the positive and negative object instances as learning samples. As per the original implementation of the R-HOG method [38], an Support Vector Machine (SVM) is used here. It turns out that the SVM learning algorithm fails to converge even after a large number of iterations. This could be attributed to the fact that the scallop profiles vary significantly based on their position in the image. To overcome this limitation, a lookup table similar to the one used to learn the scallop profiles in the classification layer is generated. The only difference here is that instead of saving a reference scallop template vector, a reference HOG vector for only positive scallop instances from the learning set is recorded. The reference HOG descriptor for a pixel coordinate in the image is taken to be the mean of all the HOG descriptors of scallop instances inside a 40×40 window around the point.

For each instance classified as a scallop from the classification layer, its HOG descriptor is compared with its corresponding learned reference HOG descriptor from the lookup table. Since HOG feature vectors are essentially histograms, the Earth Mover's Distance (EMD) metric [39] is used to measure the dissimilarity between feature and object histograms. Let A and B be two histograms, and let m and n be the number of bins in A and B, respectively. Denote d_{ij} the spatial (integer) distance between bin i in A and bin j in B, and f_{ij} the smaller number of items that can be moved between bins i and j to ultimately make both histograms match (this is known as the *optimal flow* and can be found through a process of solving a linear program [39]). Then the EMD metric D_{emd} that quantifies dissimilarity between two histograms A and B would

	HOG		WCTM	
	Dataset 1	Dataset 2	Dataset 1	Dataset 2
TP[1] from Classification Layer	183	1 759	183	1759
FP[2] from Classification Layer	7 970	52 456	7 970	52 456
TP after Layer IV	179	1 689	176	1 685
FP after Layer IV	7 752	51 329	2 924	16 407
Decrease in TP after Layer IV	4 (2.2%)	70 (4%)	7 (3.8%)	74(4.2%)
Decrease in FP after Layer IV	218 (2.7%)	1 127 (2.1%)	5 046 (63.3%)	36 049 (68.7%)

[1] TP–True Positives
[2] FP–False Positives

Table 1.4: Comparison of tested false positive filter layer methods

be expressed as

$$D_{\mathsf{emd}}(A, B) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}} \ .$$

A precision-recall curve (shown in Figure 1.13b) with the classification threshold set as 2.816 (which corresponds to 96% recall rate, same rate used to set the WCTM threshold). Any object with EMD distance value less than this threshold is considered as a scallop. Though this threshold can capture 96% of the scallops, very few false positives actually get eliminated (less than 3%).

1.7 Results

The multi-layered detection approach is tested on two separate datasets containing 1 299 and 8 049 images, respectively. Among the two candidate methods tested for the fourth layer, WCTM was chosen over HOG due to it superior performance in terms of eliminating false positives. The difference in performance between HOG and WCTM is given in Table 1.4 for both datasets. Rows 1 and 2 in Table 1.4 show the true positives and false positives, respectively, that are filtered down from the initial 3 layers (Layers I-III). With these values as baseline, the thresholds for both HOG and WCTM were chosen to retain a high recall rate of close to 96%. This ensures that very few true positives are lost and their performance is primarily assessed through the reduction in false positives (row 6 of Table 1.4).

Since the thresholds are set such that the recall rate is high in both methods, the decrease in true positives is less than 5% in both HOG and WCTM. However there is a significant reduction in false positives (63.3% for dataset 1 and 68.7% for dataset 2) due to WCTM. On the other hand, the decrease in false positives is relatively small (less than 3%) for HOG. It is not clear at this point why the HOG filter fails to remove false positives. One reason could be that the HOG filter derived from its native implementation for human detection in [38] might need further customization and even weighting through standard deviation weights like in WCTM. Further study and detailed analysis is required to investigate and possibly improve its performance. In any case, the results support the inclusion of WCTM as the false positive filter layer in the multi-layer scallop detection and counting process pipeline.

The overall performance of the four-layer pipeline is shown in Table 1.5. The results are compared to manually labeled ground truth. Only a subset of the available scallops—scallops at least 80 pixels horizontally and 60 pixels vertically away from the image boundaries—were used as ground truth. This was done to leave out scallops near the boundaries that were affected by severe vignetting effects. Such scallops were often too dark (see Figure 1.1) and very difficult to correct using standard vignetting correction algorithms. Furthermore, the scallop templates for scallops near the boundaries are such that their prime feature, the dark crescents, blend into the dark borders of the image (see Figure 1.9a). Inclusion of the boundaries would cause almost any objects near the boundary to be classified as scallops, resulting in a large number of false positives. It is also interesting to note that scallops only partially visible near the image boundaries were excluded in the manual counts performed [29].

Table 1.5 shows the results of the 3-layer pipeline along with the improvements in terms of the reduction in false positives as a result of introducing the fourth processing layer. The true positive percentages shown are computed with reference to the valid ground truth scallops (row 3 of table 1.5), i.e., scallops away from image boundaries. In dataset 1, which contains 1 299 images, the four-layer filtering results in a 70.4% overall recall rate, while in dataset 2 that contains 8 049 images the overall recall rate

Table 1.5: Results of multi-layer scallop classification

	Dataset 1	Dataset 2
Number of images	1,299	8,049
Ground Truth Scallops	363	3,698
Valid Ground Truth Scallops	250	2,781
True positives after Visual Attention Layer	231 (92.4%)	2,397 (86.2%)
True positives after Segmentation Layer	185 (74%)	1,807 (64%)
True positives after Classification Layer	183 (73%)	1,759 (63.2%)
True positives after False Positive Filter Layer	176 (70.4%)	1,685 (60.6%)
False positives after Classification Layer	7,970	52,456
False positives after False Positives Filter Layer (WCTM)	2,924	16,407
Decrease in false positives (due to WCTM)	63.3%	68.7%

is 60.6%. Though the addition of the fourth false positive layer results in a small drop of 2.6% in recall rate, it eliminates over 63% of the false positives in both datasets. There is no clear reason for the better performance of this pipeline on dataset 2 both in terms of recall rate and decrease in false positives compared to dataset 1.

1.8 Discussion

The four-layer automated scallop detection approach discussed here works on feature-poor, low-light imagery and yields overall detection rates in the range of 60–75%. Related work on scallop detection using underwater imaging [40, 10], reported higher detection rates, but the quality of the images used was visibly better. Specifically, the datasets on which the alternative algorithms [10] operated on, exhibit much more uniform lighting conditions, higher resolution, brightness, contrast, and color variance between scallops and background (see Figure 1.14). Evidence of this can be seen in Figure 1.14: the color variation between scallops and background data is reflected in the saturation histogram of Figure 1.14. While the histograms of scallop regions in the datasets of Table 1.5 is often identical to the global histogram of the image, the histograms of the Woods Hole data used by the alternative algorithms [10] present a

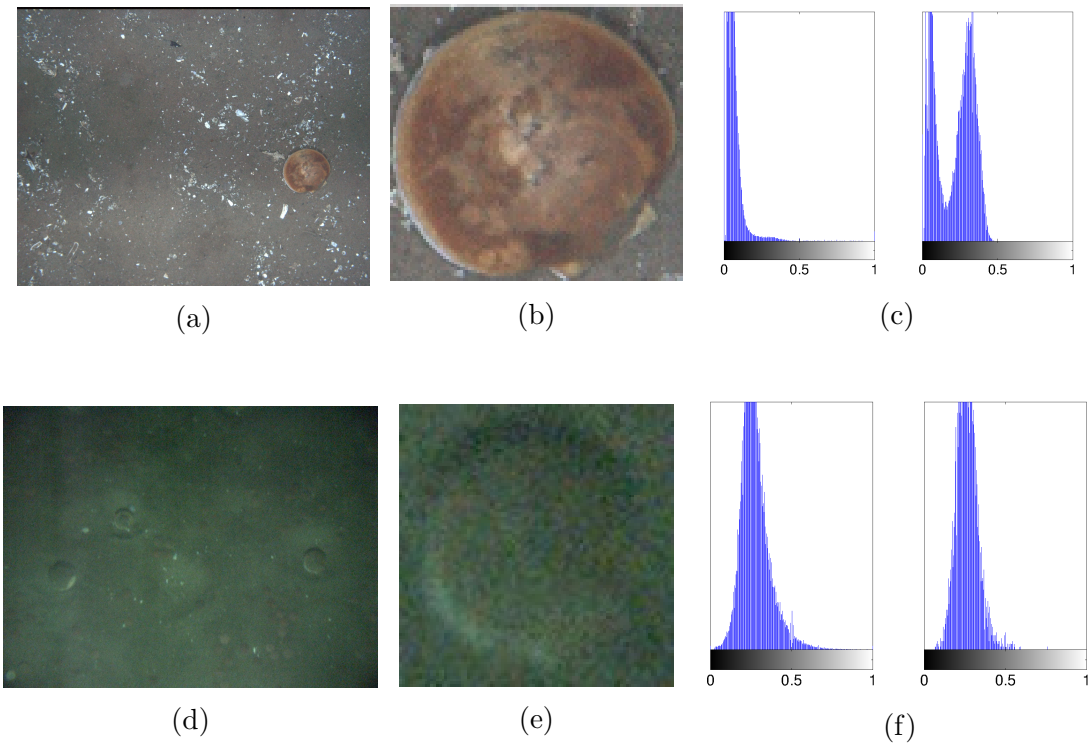
bimodal saturation histogram (Figure 1.14c), from which foreground and background are easily separable.

Compared to another alternative approach that uses a series of bounding boxes to cover the entire image [21], the one reported here employs only ten windows per image, scanning the images at a much faster rate. Additionally, the detection rates there [21] were based on a dataset of just 20 images; statistically significant differences in performance rates between that approach and the one reported here would need much larger image samples.

1.9 Conclusions and Futurework

With the increasing use of underwater robotic platforms, terrabytes of imagery datasets featuring millions of images are becoming commonplace. The current practice of manual processing of these underwater images introduces a bottleneck. In the spirit of this scallop counting work, designing better and faster automated tools to characterize animals and other natural underwater phenomenon from images is imperative for future marine environmental studies.

This work is a step toward the development of an automated procedure for scallop detection, classification and counting, based on low-resolution imagery data obtained in the organisms' natural environment. The uniqueness of the reported method lies in its ability to handle poor lighting and low-contrast imaging conditions. A large natural datasets of over 8000 images have been used to validate this four-layered framework. Augmenting a previously developed three-layer scallop counting framework with a dedicated false-positive filtering layer has a drastic effect in terms of reducing the number of false positives. The study noted that a filter based on a custom WCTM method outperforms HOG in this specific application context. The multilayer framework reported is verified to be modular, and it allows easy adaptation of different layers for various applications like counting other sea organisms. Designing such tools with further improvements in form of higher detection rates and lower false positives is required to help advance future marine animal studies.



(a) (b) (c)

(d) (e) (f)

Figure 1.14: Representative samples of different imagery data on which scallop detection algorithms may be called to operate on. Figures 1.14a and 1.14d, show an image containing a single scallop from the dataset used by Dawkins et al.[10] (used with permission from the authors) and the datasets used in this paper respectively. A magnified view of a scallop cropped from Figure 1.14a and 1.14d can be seen in Figures 1.14b and 1.14e respectively. Figure 1.14c gives the saturation histogram of background or the complete image in Figure 1.14a to left and saturation histogram of Figure 1.14b to the right. Similarly, Figure 1.14f gives the saturation histogram of Figure 1.14d to the left and saturation histogram of Figure 1.14e to the right. The bimodal nature of the scallop histogram in Figure 1.14c derived from the dataset used in Dawkins et al.[10], clearly portrays the distinguishing appearance of the scallop pixels from the rest of the image, making it easily identifiable. The datasets we used did not exhibit any such characteristics (as seen in Figure 1.14f) to aid the identification of scallops.

1.10 Future Work

One future direction would be to further reduce false positives by enhancing the false positives filter layer by using multiple scallop reference templates for each pixel location. These new templates could be designed to capture the bright crescents that sometimes appear due to the visible interior of the lower valve of a scallop when the scallop shell is partly open. As this crescent appearance is only dependent on the relative scallop orientation with respect to the camera, it can occur at any point in the periphery of a scallop. If these bright crescents were to be used in conjunction with dark crescents multiple templates will be required to model scallops at each pixel location. This idea is supported by inspection of recently collected high-resolution scallop data, which indicate additional definitive features connecting the position of the bright and dark crescents along with their relative intensities. We believe that even without major changes to the current framework, testing on higher resolution images could produce much better performance outcomes (both in terms of detection and false positive rates). The unavailability of ground truth for the new datasets makes it hard to provide evidence of any performance at this point. It is also expected that using more targeted color and light correction methods [10] as a part of image preprocessing will improve results.

Building robust object classification techniques capable of handling noisy data, is one of the primary directions where improvement is necessary. It is possible that a single noisy image of a target object might lack the information needed to accurately recognize it. With this in mind, a multi-view object recognition approach that combines information from multiple images is proposed in Chapter **??**.

Acronyms

AUV Autonomous Underwater Vehicle. 8, 14

BUVA Bottom-Up Visual Attention. 8, 20

DVL Doppler Velocity Log. 15

EMD Earth Mover's Distance. 33, 34

HOG Histogram of Gradients. 29, 31–35, 37

INS Inertial Navigation System. 15

RSA Research Set-Aside. 2, 15, 18

SVM Support Vector Machine. 33

TDVA Top-Down Visual Attention. 3, 12, 20, 22

WCTM Weighted Correlation Template Matching. 29, 31, 32, 34–37

BIBLIOGRAPHY

[1] JF Caddy. Spatial model for an exploited shellfish population, and its application to the georges bank scallop fishery. *Journal of the Fisheries Board of Canada*, 32(8):1305–1328, 1975.

[2] FM Serchuk, PW Wood, JA Posgay, and BE Brown. Assessment and status of sea scallop (placopecten magellanicus) populations off the northeast coast of the united states. In *Proceedings of the National Shellfisheries Association*, volume 69, pages 161–191, 1979.

[3] DR Hart and PJ Rago. Long-term dynamics of us atlantic sea scallop placopecten magellanicus populations. *North American Journal of Fisheries Management*, 26(2):490–501, 2006.

[4] KS Naidu and G. Robert. Fisheries sea scallop placopecten magellanicus. *Developments in Aquaculture and Fisheries Science*, 35:869–905, 2006.

[5] Fisheries of the United States. Fisheries of the United States, Silver Spring, MD. Technical report, National Marine Fisheries Service Office of Science and Technology, 2012.

[6] AA Rosenberg. Managing to the margins: the overexploitation of fisheries. *Frontiers in Ecology and the Environment*, 1(2):102–106, 2003.

[7] P. Kannappan and HG Tanner. Automated detection of scallops in their natural environment. In *IEEE Mediterranean Conference on Control & Automation, 2013*, pages 1350–1355, 2013.

[8] P. Kannappan, JH Walker, AC Trembanis, and HG Tanner. Identifying sea scallops from benthic camera images. *Limnology and Oceanography: Methods*, 12(10):680–693, 2014.

[9] P. Kannappan, HG Tanner, AC Trembanis, and JH Walker. Machine learning for detecting scallops in AUV benthic images: Targeting false positives. *Computer Vision and Pattern Recognition in Environmental Informatics*, pages 22–40, 2015.

[10] M. Dawkins, C. Stewart, S. Gallager, and A. York. Automatic scallop detection in benthic environments. In *IEEE Workshop on Applications of Computer Vision*, pages 160–167, 2013.

[11] RJ Webster. PhD thesis, 2013.

[12] FA Wichmann, J. Drewes, P. Rosas, and KR Gegenfurtner. Animal detection in natural scenes: critical features revisited. *Journal of Vision*, 10(4):6–6, 2010.

[13] C. McGavigan. A quantitative method for sampling littoral zooplankton in lakes: The active tube. *Limnology and Oceanography: Methods*, 10:289–295, 2012.

[14] CP Stelzer. Automated system for sampling, counting, and biological analysis of rotifer populations. *Limnology and Oceanography: Methods*, 7:856, 2009.

[15] AL Forrest, ME Wittmann, V. Schmidt, NA Raineault, A. Hamilton, W. Pike, SG Schladow, JE Reuter, BE Laval, and AC Trembanis. Quantitative assessment of invasive species in lacustrine environments through benthic imagery analysis. *Limnology and Oceanography: Methods*, 10:65–74, 2012.

[16] T. Schoening. *Automated detection in benthic images for megafauna classification and marine resource exploration: supervised and unsupervised methods for classification and regression tasks in benthic images with efficient integration of expert knowledge*. PhD thesis, Universität Bielefeld, 2015.

[17] C. Spampinato, YH Chen-Burger, G. Nadarajan, and RB Fisher. Detecting, tracking and counting fish in low quality unconstrained underwater videos. In *3rd International Conference on Computer Vision Theory and Applications*, pages 514–519. Citeseer, 2008.

[18] DR Edgington, DE Cline, D. Davis, I. Kerkez, and J. Mariette. Detecting, tracking and classifying animals in underwater video. In *Oceans'06 MTS/IEEE-Boston Conference and Exhibition*, pages 1–5. IEEE, 2006.

[19] RN Williams, TJ Lambert, AF Kelsall, and T. Pauly. Detecting marine animals in underwater video: Let's start with salmon. In *Americas Conference on Information Systems*, volume 1, pages 1482–1490, 2006.

[20] B. Zion. The use of computer vision technologies in aquaculturea review. *Computers and Electronics in Agriculture*, 88:125–132, 2012.

[21] EO Guòmundsson. Detecting scallops in images from an auv. Master's thesis, University of Iceland, 2012.

[22] K. Enomoto, M. Toda, and Yasuhiro Kuwahara. Scallop detection from sand-seabed images for fishery investigation. In *2nd International Congress on Image and Signal Processing*, pages 1–5. IEEE, 2009.

[23] K. Enomoto, M. Toda, and Y. Kuwahara. Extraction method of scallop area in gravel seabed images for fishery investigation. *IEICE Transactions on Information and Systems*, 93(7):1754–1760, 2010.

[24] R. Fearn, R. Williams, M. Cameron-Jones, J. Harrington, and J. Semmens. Automated intelligent abundance analysis of scallop survey video footage. *AI 2007: Advances in Artificial Intelligence*, pages 549–558, 2007.

[25] National Marine Fisheries Service Northeast Fisheries Science Center (NEFSC). 50th northeast regional stock assessment workshop (50th SAW) assessment report. Technical Report 10-17, US Dept Commerce, Northeast Fisheries Science Center, 2010.

[26] SR Jenkins, BD Beukers-Stewart, and AR Brand. Impact of scallop dredging on benthic megafauna: a comparison of damage levels in captured and non-captured organisms. *Marine Ecology Progress Series*, 215:297–301, 2001.

[27] GE Rosenkranz, SM Gallager, RW Shepard, and M. Blakeslee. Development of a high-speed, megapixel benthic imaging system for coastal fisheries research in alaska. *Fisheries Research*, 92(2):340–344, 2008.

[28] SM Gallager, H. Singh, S. Tiwari, J. Howland, P. Rago, W. Overholtz, R. Taylor, and N. Vine. High resolution underwater imaging and image processing for identifying essential fish habitat. In D.A. Somerton and C.T. Glentdill, editors, *Report of the National Marine Fisheries Service Workshop on Underwater Video analysis*, NOAA Technical Memorandum NMFS-F/SPO-68, pages 44–54. 2005.

[29] JH Walker. Abundance and size of the sea scallop population in the mid-atlantic bight. Master's thesis, University of Delaware, 2013.

[30] L. Oremland, D. Hart, L. Jacobson, S. Gallager, A. York, R. Taylor, and N. Vine. Sea scallop surveys in the 21st century: Could advanced optical technologies ultimately replace the dredge-based survey? Presentation made to the NOAA Office of Science and Technology, October 2008.

[31] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4(4):219–227, 1985.

[32] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998.

[33] D. Walther and C. Koch. Modeling attention to salient proto-objects. *Neural Networks*, 19(9):1395–1407, 2006.

[34] V. Navalpakkam and L. Itti. An integrated model of top-down and bottom-up attention for optimizing detection speed. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2049–2056. IEEE, 2006.

[35] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[36] G. Taubin. Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(11):1115–1138, 1991.

[37] N. Chernov. *Circular and linear regression: Fitting circles and lines by least squares*. Taylor and Francis, 2010.

[38] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893. IEEE, 2005.

[39] Y. Rubner, C. Tomasi, and LJ Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[40] M. Dawkins. Scallop detection in multiple maritime environments. Master's thesis, Rensselaer Polytechnic Institute, 2011.