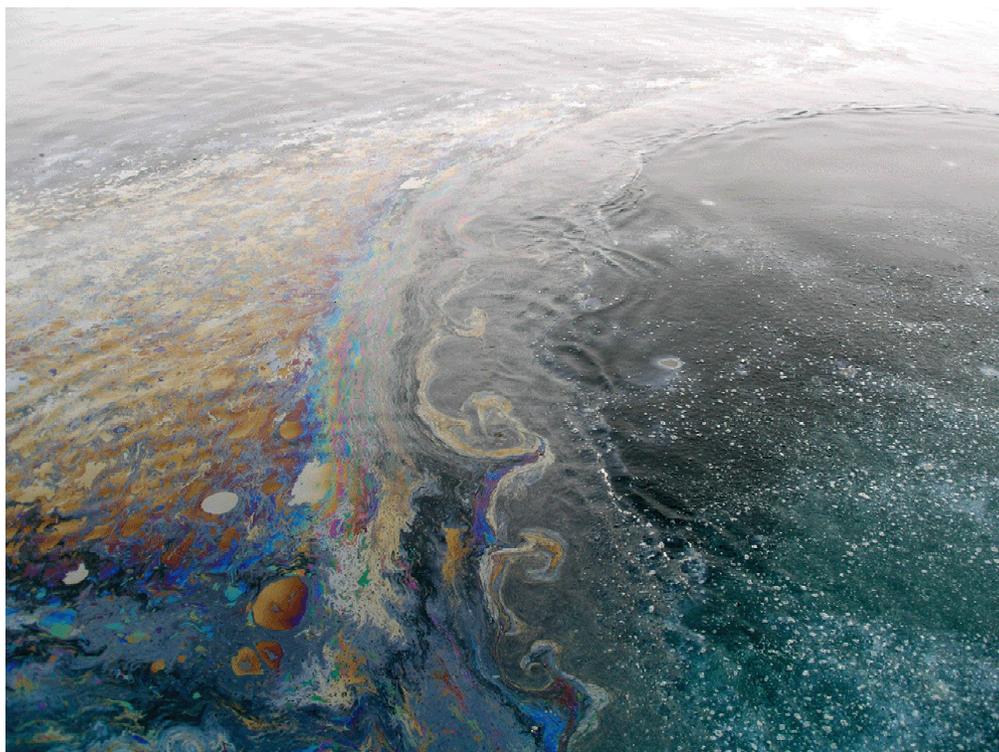


**DEVELOPMENT AND EVALUATION OF A COST-EFFECTIVE AERIAL
IMAGING SYSTEM FOR OIL SPILL AND COASTAL IMPACT MONITORING.**

Final Report
Submitted to the California Dept. of Fish and Game
Office of Oil Spill Prevention and Response
Sacramento, CA
December 11th, 2007

Contract No. P0375037



Principal Investigator: Dr. Jan Svejksky, Ocean Imaging Corp. 201 Lomas Santa Fe Dr.,
Suite 370, Solana Beach, CA 92075 (858) 792-8529, Fax: (858) 792-8761, jan@oceani.com

OSPR Sponsor: Judd Muskat, CDFG Office of Oil Spill Prevention and Response,
1700 "K" Street, Sacramento, CA 95814 (916) 324-3411, JMuskat@dfg.ca.gov

Disclaimer

This report has been reviewed by the California Dept. of Fish and Game's (CDFG) OSPR staff for technical adequacy according to contractual specifications. The opinions, conclusions, and recommendations contained in the report are those of the author and do not necessarily reflect the views and policies of CDFG. The mention of a trade name or any commercial product in the report does not constitute an endorsement or recommendation for use by CDFG.

On the Cover: Edge of an active oil and gas seep in the Santa Barbara Channel, California photographed from a field sampling vessel

EXECUTIVE SUMMARY

This project aimed to develop and test multispectral oil spill impact detection methods for ocean and terrestrial use, and evaluate the efficiency and potential usefulness to OSPR of a 4-channel, portable UV-Vis-nearIR sensor for oil spill mapping, support of recovery operations and monitoring environmental damage and remediation success. The research was done during 2004 through 2006.

For work targeting the detection and mapping of at-sea oil spills, Santa Barbara Channel oil seeps were used to obtain imagery over oil slicks and other possible false targets e.g. kelp, suspended sediment, wind/wave effects and sunglint. Two flights were done to test different 4-channel wavelength combinations for best discrimination ability between oil and false targets. Several multispectral classification algorithms were tested. Results indicated that good oil identification success is possible but some false targets (mostly from sunglint areas) could not be eliminated. The final approach was to develop an application specific neural network-based algorithm that incorporates both spectral and spatial pattern variables. This method results in minimal false targets and can even provide some information on oil thickness distribution patterns. This approach was tested over the Santa Barbara seeps using simultaneous aerial and ship-based data sampling.

For terrestrial oil spill work, the original proposal intended to utilize long-term contaminated Bolsa Chica wetlands as the testing grounds. Aerial data were collected but, despite repeated requests, ground truth data compiled by US Fish & Wildlife Service could not be obtained. A real spill incident that occurred during the project was instead incorporated into the work. On 4/27/2004 a diesel pipeline fractured in Suisun Marsh east of San Francisco Bay and contaminated a vast area of the marsh. 2 imaging overflights were performed (funded separately from this project) with a time separation of 2 months, and change detection methodology was developed to accurately map changes in ground cover type (e.g. vegetated turned bare) and to measure changes in growth vigor. The developed techniques were successfully used to provide OSPR and other agency and corporate entities involved in the case with high resolution maps of environmental change over the spill-affected area.

The project proved that the development and operational utilization of a portable multispectral imaging system for oil spill mapping is very feasible and could provide major improvements in oil spill response. The developed system, presently based around Ocean Imaging's DMSC-MKII aerial imager, is available for operational use, and has been utilized by CDFG/OSPR in other spill cases since the project's termination (e.g. Romic chemical spill in 2006).

1. PROJECT BACKGROUND

Satellite and aerial imaging can, in principle, provide a convenient means to detect and precisely map marine and terrestrial oil spills and seeps, and to monitor the environmental effects of oil impacted soils and vegetation. Significant advances have been made (primarily in Europe and Canada) in oil spill detection capabilities by UV/Vis/IR sensors and Synthetic Aperture Radars (SARs). Although utilized to aid oil spill recognition in Europe, satellite sensors have a number of disadvantages for their routine use in spill monitoring. These include low revisit frequency (2-6x/month), relatively low spatial resolution, high cost, and high false target occurrence. Aerial sensors offer the advantages of on-call availability, very high resolution, and various enhancements to minimize false targets. Very few aircraft SARs exist, however, and their use is extremely expensive. Likewise, most UV/Vis/IR sensors that have been shown to effectively detect oil-impacted land surfaces are hyperspectral instruments which are expensive to operate and often difficult to mobilize and use. Our premise in this project was that full hyperspectral capabilities are not required for efficient oil impact detection. A cost-effective, easy-to-mobilize aerial sensor offering 4 wavelength channels spanning the UV/Vis/NIR range can be configured to provide adequate data for oil detection to support a variety of OSPR's activities.

2. PROJECT OBJECTIVES

This project aimed to develop and test multispectral oil spill impact detection methods for ocean and terrestrial use, and evaluate the efficiency and potential usefulness to OSPR of a 4-channel, portable UV-Vis-nearIR sensor for oil spill mapping, support of recovery operations and monitoring environmental damage and remediation success.

The work utilized Ocean Imaging's (OI's) existing instrument as the imaging platform. OI owns and operates the DMSC-MKII aerial imager, manufactured by SpecTerra Ltd. in Australia. The DMSC is a portable, 4-channel sensor with 12-bit radiometric resolution. It's channels can be customized for any wavelength between approximately 400nm (UV) and 950nm (near-IR).

Simplicity, portability and ease-of-use under real-world, operational conditions were paramount in guiding the development phases of this project.

3. RESULTS

3.1 At-sea Oil Spill Mapping Algorithm

This project's premise for developing an oil-on-water mapping algorithm using multispectral imagery was that it be as universal as possible to allow operational use in different geographic regions and under different oceanic and atmospheric conditions.

Natural oil seeps in the Santa Barbara Channel, California were chosen as the target for algorithm development and testing because the area contains not only oil slicks with varied thickness and weathering characteristics but also abundant potential false targets such as floating kelp and turbidity patterns. During the initial overflight mission, imagery was collected with several different filter combinations on the DMSC's four channels, spanning the UV – visible – nearIR range. The purpose of that work was to determine the most effective 4 wavelength combination for reliably distinguishing floating oil and separating its reflectance characteristics from other false targets. Our results indicated that unweathered and weathered oil can be effectively distinguished from background water of varying turbidity by applying maximum likelihood classifiers to multispectral imagery that contains at least one band in the 400-450nm range, one band in the 500-550nm range and one band within 625-675nm. A near-IR band, 750-850nm, was highly effective in distinguishing any floating algae, e.g. kelp, from the actual oil slicks, as is shown in Figure 1.

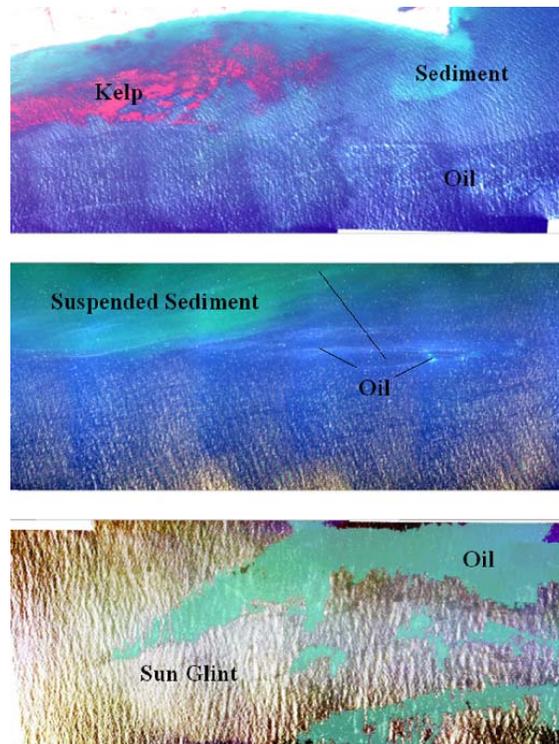


Figure 1. Multispectral imagery off Coal Point, California showing oil slicks and various potential artifacts.

Further analysis of the data revealed, however, that wind and wave effects coupled with sun glint often create patterns that cannot be reliably separated from oil signatures using multispectral classification algorithms that rely solely on spectral reflectance relationships. Additional imagery collected during a subsequent overflight mission also revealed that background reflectance variability due to variability in overall water color and sun angle preclude the utilization of a general or master classification parameter set that could be universally applied to any future image data.

Realizing the above limitations, our final oil mapping algorithm approach utilized a neural network-based classifier which considers not only multispectral relationships but also textural information within the image. The neural network (NN) approach greatly improved the elimination of false targets due to sunglint and wave patterns. A comparative example is shown in Figure 2 and details of running the procedure are in Appendix A.

It is important to note that applying the NN oil classifier to an image data set first involves operator-assisted training of the network. Although this is generally a relatively rapid process (5-15 minutes), it prevents the mapping system from creating a fully automated (i.e. without any operator intervention) oil map product. Our research in this and subsequent projects focusing on mapping oil-on-water have shown that this semi-automated procedure is preferable to a fully automated one with the imaging hardware and software resources presently available. Tests with a fully automated algorithm version, relying on stored oil reflectance characteristics, was deemed too error-prone to be of satisfactory operational value under varied oceanographic and atmospheric conditions.

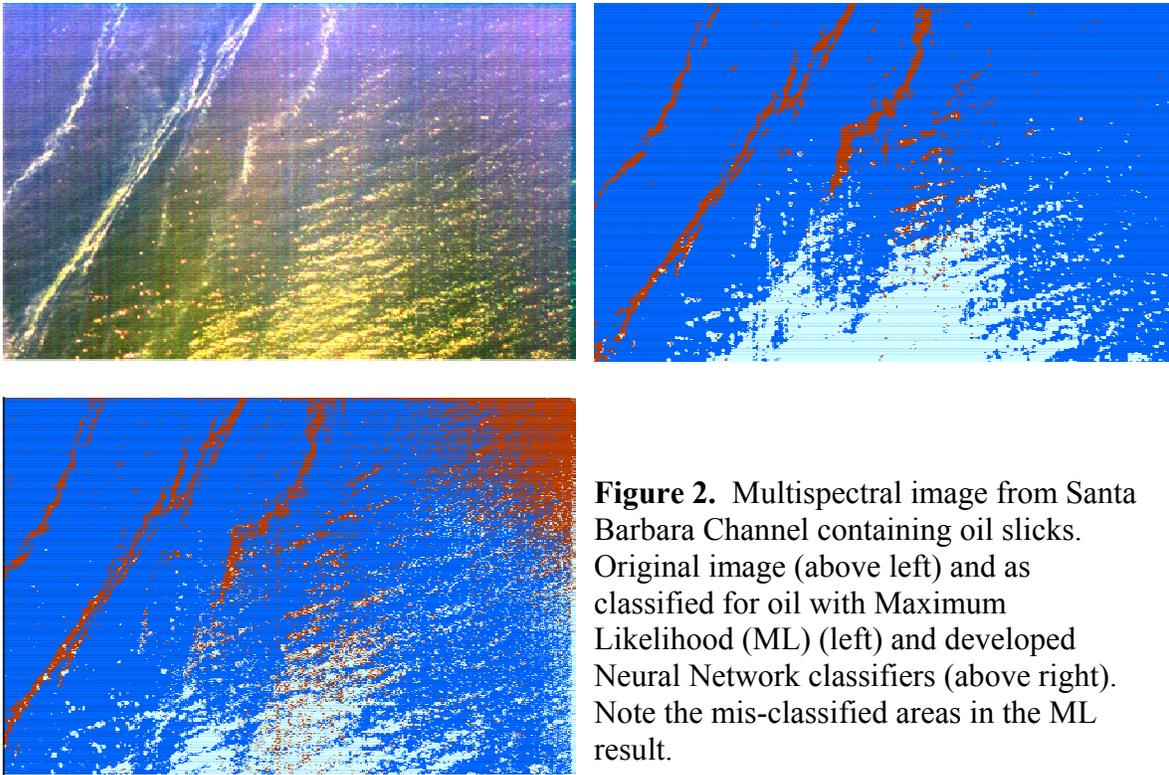


Figure 2. Multispectral image from Santa Barbara Channel containing oil slicks. Original image (above left) and as classified for oil with Maximum Likelihood (ML) (left) and developed Neural Network classifiers (above right). Note the mis-classified areas in the ML result.

The accuracy of the developed oil mapping algorithm was tested during subsequent overflights and simultaneous boat sampling done in the Santa Barbara Channel for this project, as well as for a subsequent project funded by the Minerals Management Service (MMS) to enhance the oil mapping capabilities developed here by adding actual oil film

thickness estimates. The system's oil mapping accuracy will vary for each data set due to oceanic and atmospheric conditions, as well as the distribution patterns of the oil itself (e.g. long thin strands vs. a more coherent slick). Our initial evaluation tests in the Santa Barbara Channel indicate a very high degree of accuracy for oil films thicker than approximately 0.02mm. The final version of the algorithm tends to underclassify sheens (very thin sheens share the characteristics of sunglint areas) but we deemed this preferable (under operational spill response conditions) to better sheen discrimination at the expense of significant increase in false targets. We are continuing to quantitatively assess the algorithm's accuracy in subsequent projects.

3.2 Terrestrial Oil Spill Mapping Technology Development

This project's development of oil spill sensing technologies for land use primarily focused on detecting and mapping the oil's impact on soils and vegetated surfaces. As was already noted, the originally proposed study area was the Bolsa Chica marsh in Orange County, California. The marsh has been the site of numerous petroleum spills through the years and at the time of this project was slated for major restoration. Although imagery was collected of the entire study area, analysis of the data and development of impact-detection/mapping techniques required field data for the development and validation. Such a field data set containing contamination measurements at thousands of points throughout the marsh was to be made available to us by the U.S. Fish & Wildlife Service who directs the restoration effort. Despite multiple requests for the data, OI was not able to obtain it and thus did not utilize the Bolsa Chica region for further work.

On 4/27/2004 a pipeline carrying diesel fuel through the Suisun Marsh, California fractured and spilled the fuel into a network of sloughs where it was further spread by tidal flooding. The CDFG subsequently contacted OI to inquire if a remote sensing-based survey could be done that would aid in delineating and assessing environmental damage related to the spill. Utilizing its former plans for research in Bolsa Chica as well as other wetlands mapping experience, OI suggested a change detection approach based on algorithm-detected changes in ground cover type and plant vigor between two image sets collected at different times after the spill. The two flights were done on 5/18/04 and 7/15/04 (all extra costs were funded outside of CDFG).

OI concentrated on identifying three prime changes: 1) areas which were flooded in May but had no standing water in July (due to the draining of the marsh after the spill); 2) areas which were vegetated in May but turned to bare soil by July; 3) areas which experienced significantly decreased chlorophyll vigor (as measured by a change in the Normalized Difference Vegetation Index (NDVI) computed from the multispectral data) between May and July.

The two data sets were first processed to identify standing water areas in the May imagery, and areas of change due to vegetated ground being denuded or, conversely, bare ground becoming vegetated between May and July. This was done using a customized

algorithm developed by OI in previous wetland mapping projects. The two data sets were then also used to compute an NDVI May-July difference.

An assumption was made that any areas which showed increased or unchanged vigor or plant density were not significantly affected by the incident. These areas were therefore masked and eliminated from further analysis.

An analysis product showing all areas of significantly reduced vegetation vigor was then manually edited to eliminate areas where the decrease in vigor could obviously be ascribed to natural/seasonal causes. This filtering was guided by considering the trends of various differentiated plant types in the northern control region away from the initially flooded features. However, since the draining of both the affected and control regions caused the death of any aquatic plants and algae and possibly caused lack-of-water stress on plants in and immediately around the flooded regions, such changes were retained in the final analysis product.

Because OI was not provided with any quantitative ground truth data to use for algorithm training or validation purposes, the analyses represent objective depictions of vegetation-related change in the sense that OI staff made no assumptions about the locations or type of change caused by direct petroleum contact, and made no attempts at specifically isolating it. OI was furnished with 8 field photos by OSPR, taken on 5/3/04, 5/6/04 and 5/12. Each photo contained the date, direction of view and location code corresponding to the Cleanup Segment Map. These photos were helpful in identifying some types of vegetation and several appeared to show good correspondence with the vegetation change analyses results.

Because the spill case has not yet been closed, we cannot provide any detailed information related to the environmental damage assessments derived from the developed analysis approach. As can be seen from examples in Figure 3, however, the remotely sensed change detection analysis provided highly detailed vegetation and land cover change mapping capabilities. The technique was used again after a chemical spill over Laumeister Marsh, California in 2006, where the change detection results were statistically compared to a control marsh area and the conclusions were corroborated by detailed field sampling.

In addition to providing more comprehensive synoptic information on the possible effects of petroleum or chemicals spilled on or into a land area, the developed techniques provide image and map layer data products suitable for direct input into Geographical Information Systems (GIS). This feature allows, in turn, further quantitative assessments such as the rapid computation of the total area of vegetation affected.

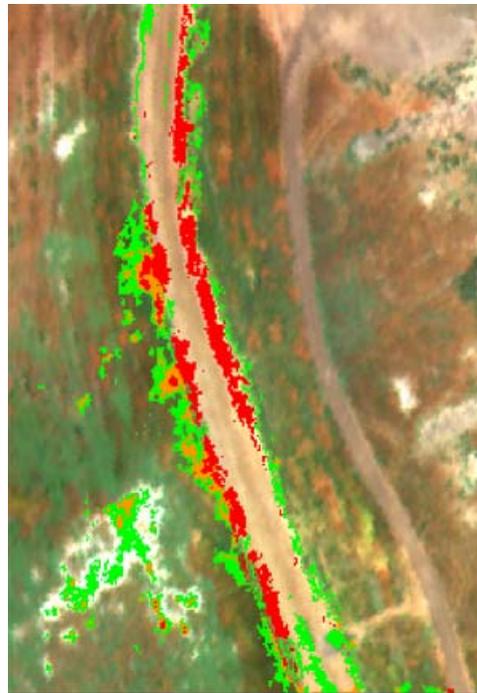


Figure 3. Field photo of an edge of one of the diesel-affected sloughs in Suisun Marsh (above) and same area processed for vegetation change from aerial multispectral imagery (right). Note the correspondence of vegetation dieoff features (red) to the band of coated vegetation seen in the field photo immediately after the spill.

4. OPERATIONAL PRACTICALITY OF THE DEVELOPED SYSTEM

A pseudo-operational oil seep mapping was done in October, 2005 over the Santa Barbara Channel during experiments that included ship-based ground truthing for a related MMS-funded project targeting oil slick thickness algorithm development. Several oil slick targets were imaged, one of which is shown in Figure 4. Mounting of the sensor in the aircraft (CDFG's Partenavia) took 15 minutes with the mounting plate and power supply already available or approximately 1 hour if the plate and power connections would have to be newly installed. At altitude of 7000 feet, providing 1m ground data resolution 3km²/minute could be imaged (at 100knts ground speed).

Subsequent data processing times depend, logically, on the overall area of the spill. Spills up to several km² will be contained within a few image frames at typical altitudes and processing, including mosaicking the frames into one image and running the NN classifier takes 15-30 minutes. Longer times must be expected for larger spills or spills where the oil has separated into multiple areas that must be imaged and processed separately. Although we did not have the opportunity to test direct plane-to-land image transfers during the test overflight, air-to-land transfers were successfully tested during more recent follow-on work using a commercially available wireless data network.

For land impact mapping cases the same parameters apply for data collection, although ground resolution should be increased to 30cm-1m, resulting in longer overflights and longer processing times. We did not have an opportunity to image spilled oil on land

directly (in the Suisun Marsh case practically all the diesel had evaporated or was recovered before the first overflight). However, we believe the NN algorithm could be effectively utilized in cases where crude or fuel oil directly covered vegetated or bare soil areas. The change detection approach used during the project requires considerably more processing and analysis but is not, by its very nature, as time sensitive as initial response mapping.

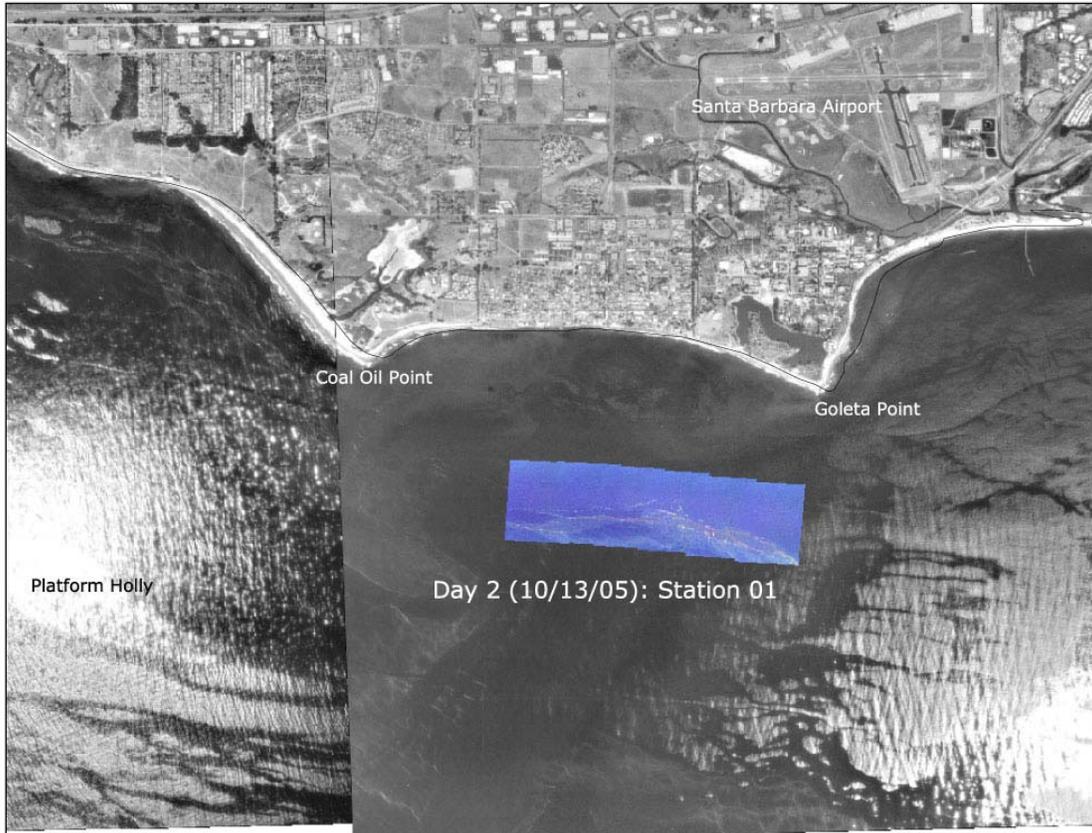


Figure 4. Oil slick target utilized in final testing off Coal Oil Point, California.

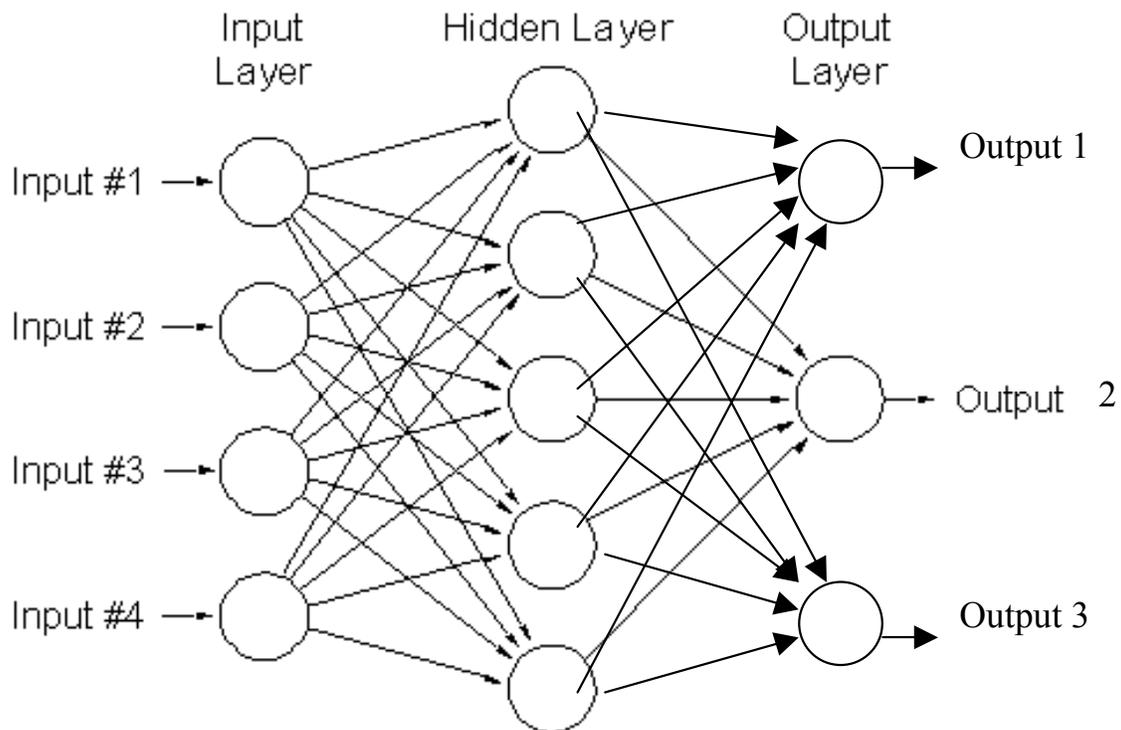
APPENDIX A

Neural Network Oil Mapping Algorithm Software Details

Neural Network Classifications in OI Imaging Tools

A Simplified Explanation of an Artificial Neural Network:

An Artificial Neural Network (ANN) is a supervised, learning classifier. An ANN can be graphically represented (below) as a graph composed of nodes (the circles) and edges (the lines connecting two nodes). An ANN is typically composed of an input node layer, one or two hidden node layers, and an output node layer. The ANN pictured below could be used to classify a pixel from a 4-band image into one of three output classes. First the band DNs would be put into the input layer (input 1 = band 1 DN, input 2 = band 2 DN, etc.). (Input values are typically scaled to the range [0,1]). Now that the values of the input layer nodes are known, the values of the hidden layer nodes can be calculated. In our case, each hidden node is connected to every input node by an edge (line). Each of these edges has a numerical weight associated with it. A weighted sum is calculated for the hidden node by multiplying each input node's value by the appropriate weight and adding them all together. This weighted sum is then entered into a special function (sigmoid function) to calculate the value of the hidden node. (This value also varies from 0 to 1). That process is repeated for each node in the hidden layer, then each node in the output layer. The output of the neural network below is 3 numerical values (one from each output node) ranging from 0 to 1. In an image classification, each of these output nodes represents a specific information class (i.e. water, grass, road). For a properly trained neural network, an input pixel of class "road" should produce ~0 for the water and grass outputs and ~1 for the road output.



Neural Network Training

As with all supervised classification methods, before a neural network can correctly classify an image, it must be trained with a set of pixels with known correct classes. Initially, all the weights of the neural network are initialized as random numbers. Next, a small subset (about 10%) of the samples from the training set will be set aside as a testing set.

The samples from the training set will then be run through the network one at a time. The output class generated for the training sample will likely be wrong at this point, so it is compared to the known correct class for the sample. The neural network then adjusts its weights slightly, so that the next time this training sample is presented to the network, it will be closer to the correct classification. (This is how the “learning” takes place). This adjustment procedure is repeated for every sample in the training set. Next, the accuracy of the network will be tested with the testing set we set aside earlier. Since the neural network has not seen these samples before, it is a good test of accuracy. The neural network DOES NOT ADJUST ITS WEIGHTS IN RESPONSE TO THE TESTING SET, it is for accuracy assessment only.

This process is called an epoch, and it will now be repeated several hundred or thousand times. Each training element will be passed through the network, the weights will be adjusted for better classification accuracy, and finally the testing set will check the accuracy at the end of the epoch.

There are two stopping conditions that can end the training of the network. The first is a set number of epochs (i.e. the network will stop after a maximum of 1000 epochs). Also, a testing set accuracy threshold can stop the training once a desired level of accuracy is obtained.

Performing a Neural Network Classification in OI Imaging Tools

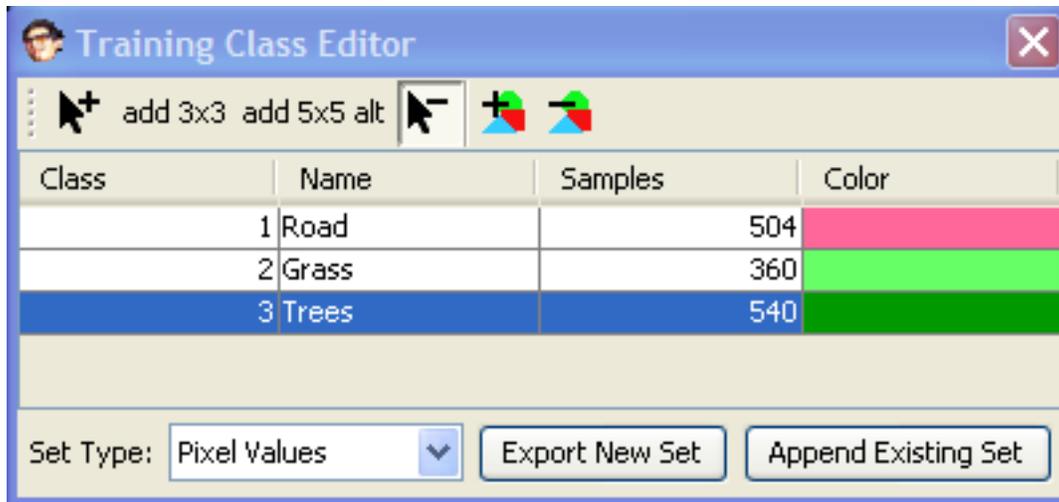
Creating a Training Set

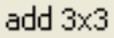
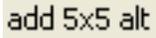
Training Sets are created just as they are for any type of supervised classification. However, there are a few tips for creating a training set that will be optimal for a neural network.

To create a training set:

1. Open the image to be used for generating the training set.
2. Select *Classification -> Create Training Set* from the main menu. The Training Class Editor Dialog will appear. Note: this dialog is only intended to be used with the image that was selected while the dialog was opened.

3. Click the Add Class Button  repeatedly until the desired number of classes has been added. (Classes may be deleted from the class table by selecting the row in the table and clicking the Delete Class Button .)
4. Click on the cells of the class table to change the class names and colors.
5. Select the class you would like to add samples to by clicking its row in the class table so it is highlighted blue.



6. Add samples (pixels) to the selected class with the assorted tools:
 - a. Add Pixel Tool  adds one pixel at a time with single clicks on the image, or a group of pixels by holding down the mouse button and drawing a rectangle around the desired pixels.
 - b. Add 3x3 Tool  adds a 3x3 square of pixels (9 pixels) centered on the pixel clicked.
 - c. Add 5x5 Alt Tool  adds a 5x5 square of pixels alternating in a checkerboard pattern (12 or 13 pixels) centered on the pixel clicked.
 - d. Delete Pixel Tool  deletes one pixel at a time with single clicks on the image, or a group of pixels by holding down the mouse button and drawing a rectangle around the pixels to be deleted.
7. Once the training set is complete, it must be saved to file. For a neural network training set, the “Set Type:” drop-down list **MUST BE SET TO “Pixel Values”**. This training set can be saved as an individual training set by clicking the “Export New Set” button, or it may be appended to an existing Pixel Values training set file by clicking the “Append Existing Set” button. When a large area composed of many image frames is being classified, it is advisable to create training sets from several sample images and combine them into one training set by appending them to each other. When appending a training set a dialog will

appear to match up the class names and numbers of the new training set with the class names and numbers of the old training set.

IMPORTANT HINTS for creating a good neural network training set:

- Training sets do not need to be extremely large, 500 to 1000 well-chosen pixels should be sufficient for each class. (If multiple images are being used to generate training sets, a few hundred samples per class per image should be sufficient). Larger training sets may increase the training time of the network without necessarily increasing the accuracy of the network.
- Training sets should reflect the variation in the class. For a given class, a group of 50 nearly identical pixels is far less useful to the neural network than a group of 20 diverse pixels of that class.
- Because of the two previous factors, a good strategy for creating a training set is to add small clusters (10-20 pixels) of sample pixels scattered throughout the image. This can correct for factors such as lighting variation across an image, and tends to get a more representative sample of the class.
- To quickly add small clusters of pixels, the 3x3 and 5x5 alt tools are very useful since they create a small cluster of sample pixels with a single click.

Training a Neural Network

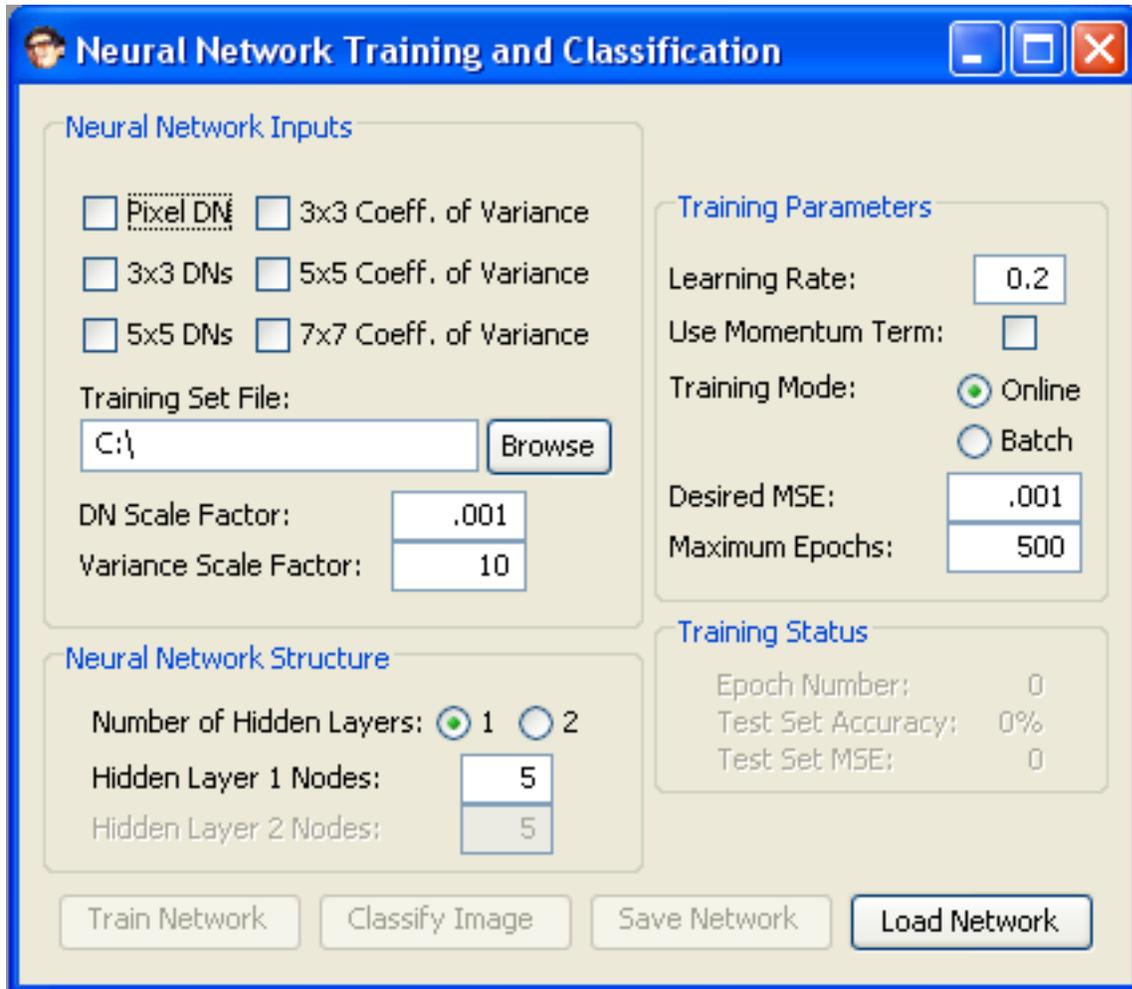
A Neural Network has many parameters to adjust the training (learning) phase. It is difficult to give general rules for these settings that work well in all situations. Experimentation will be needed to find optimum settings. However, once optimum settings have been found, they will typically work well on similar projects. For instance, the settings successfully used to train a network to classify a coastal wetland will likely work well in training a network to classify a different coastal wetland. The default settings have been found to work well in image classifications and should generally be used as the starting point.

Neural Network Inputs

The checkboxes list the possible inputs for the neural network, in reference to a pixel in a training set. These are also the values that will be considered for each pixel while classifying an image.

The most basic input is “Pixel DN”, this is the digital number of the training pixel itself. “3x3 DNs” and “5x5 DNs” include the DNs of the neighborhood around the pixel. This allows the Neural Network to consider the context of the training pixel. Only one of these options should be checked, or the Neural Network will be considering redundant information (i.e. the pixel DN is included in the 3x3 DNs). NOTE: There is currently a bug in the processing for the “5x5 DNs” so this option should not currently be used.

Other potential inputs are the coefficients of variance for 3 different window sizes. The coefficient of variance is the standard deviation of the DNs over the window divided by the mean DN Value over the window. This is intended as a textural measure. Relatively homogenous areas will have a low coefficient of variance, while highly heterogeneous areas will have a high coefficient of variance.



Recommendations: In general the simplest inputs that can satisfactorily classify the image should be used. First try simply the pixel DN. this requires very little calculation during training and classification. Next try 3x3 DNs. This requires more (9 times more) calculations than the single DN, but is often very effective. If this is not satisfactory, experiment by adding a combination of the coefficients of variance. Note that calculating coefficients of variance is very computationally intense and may greatly increase training and classification time, therefore it is not recommended unless it appears absolutely necessary.

Training Set File: The training set file to be used in creating this neural network.

DN Scale Factor: For the neural network to function properly, it is important that nearly all the input values fall into the range of 0 to 1. The “DN Scale Factor” is the number

multiplied by the DNs to convert them to the 0 to 1 range. For example if pixel DNs in an image max out at 1000, the DN Scale Factor should be 0.001, if DNs range from 0 to 4000, the DN Scale Factor should be 0.00025.

Variance Scale Factor: For the neural network to function properly, it is important that nearly all the input values fall into the range of 0 to 1. The “Variance Scale Factor” is the number multiplied by the Coefficients of Variance to convert them to the 0 to 1 range. (The coefficient of variance is defined as the standard deviation of a group divided by the groups mean value).

Neural Network Structure

The structure of a neural network is very important in determining the networks abilities. Again the general rule is to use the simplest network that can do the job. A more complex network can more easily learn a training set, but it is prone to *overfitting* or *overtraining*. Overfitting is when the network has the ability to simply memorize the training set examples without actually learning the general rules that separate the classes. This prevents the network for being able to generalize and correctly classify samples that aren't exactly the same as samples it has seen before. An overfitted network will have very high accuracy on the training set but will fail to accurately classify an entire image with pixels slightly dissimilar to those in the training set.

The power of the neural network is its ability to generalize. That is, it can learn the general rules of classification without just memorizing training samples. Simpler networks do not have the capacity to memorize a large training set and are therefore forced to generalize and less likely to overfit.

It is difficult to give a specific configuration for the network that will work in all situations. The user will initially need to experiment to find the best configuration for a given situation (i.e. wetland classification). Once a successful configuration is found, it will typically work well on other future classifications of that land type.

Recommendations:

Number of Hidden Layers: For nearly all image classifications, 1 hidden layer is sufficient. Two hidden layers should be used only if a 1 hidden layer network is unsuccessful in many attempts with different settings (i.e. number of hidden nodes, learning rate, number of epochs, a different training set). Adding a second hidden layer will increase classification time, and could lead to overfitting.

Number of Hidden Nodes: The number of hidden nodes should typically vary between 2 and 10. Again the minimum effective number of hidden nodes should be used. A higher number of nodes increases classification time and can lead to overfitting.

Training Parameters

As a network is trained, its testing error should decrease constantly until it plateaus at a specific error level. This is known as the network *converging*. Ideally the network will converge at a very low error level, however sometimes the network will become “stuck” at a high error level. If this happens, the network can be retrained, or the learning parameters may be adjusted. Each time the network is trained, the network weights are initialized with random numbers, so even with the same training parameters, the results of the network may vary. Learning parameters are best determined through experimentation with a sample of the data to be classified.

Learning Rate: When a training sample is passed through the network, the actual output is compared to the desired output (correct classification). The weights of the network are then changed so that the next time this sample is presented the networks result will be closer to correct. The learning rate determines how much the network should be changed for each sample. This value can range from 0.01 and 0.75. Higher values allow more dramatic changes to the network, but may cause the learning to be “jumpy” or “unstable”. It is best to use a small learning rate (around 0.2) so that the network can smoothly converge on a good solution.

Use Momentum Term: If the momentum term is included, when the network adjusts for a training sample, it adds a little bit of the adjustment direction from the previous training sample. This can be useful if the network in stopping the network from getting “stuck”. My experiments have shown that the momentum term is usually unnecessary and should generally NOT be used.

Training Mode: The default training mode is “Online”. In this case the weights of the network are adjusted after each training sample. If a training set has 1000 samples, this means the network weights will be adjusted 1000 times during each epoch.

In “Batch Mode” the needed weight adjustments are postponed to the end of the epoch, where they are averaged and applied all at once. In this instance weights are only adjusted once per epoch.

My experiments have shown that “Online” training tends to be faster and more reliable for image classification.

Desired MSE (Mean Squared Error): After each testing sample is put through the network, the output value of each node is compared to the desired output value. The sum of the square of these differences is the mean squared error. This setting allows the user to set an early stopping point for the training if the MSE drops below a threshold. The lower limit of the MSE varies greatly between training sets and typically needs to be determined experimentally.

Maximum Epochs: Training can be set to end after a specific number of epochs. Typically a network will need from a few hundred to a few thousand epochs to converge on a solution.

Training Status

Displays information about the training process of the neural network.

Epoch Number: The number of the current training epoch.

Test Set Accuracy: The neural networks classification accuracy of the testing set (which is a subset of the training set that was set aside for testing only). Remember that for a sample to classify as a specific class, the output node corresponding to that class should be equal to 1, while the other output nodes should be close to 0. It is rare that any samples come out so cleanly, so a pixel must only cross a certain threshold to be declared a member of a specific class. For example, a pixel representing oil with a water output value of 0.2 and an oil output value of 0.85 would be correctly classified as oil.

Test Set MSE (Mean Squared Error): After each testing sample is put through the network, the output value of each node is compared to the desired output value. The sum of the square of these differences is the mean squared error. Note: in the example given in the “Test Set Accuracy” section above, the pixel would be correctly classified as oil however since the output values are not 0 and 1, this example would have a MSE of 0.03125. Because of this, a test set can display 100% classification accuracy even as the MSE continues to fluctuate.

Dialog Buttons:

Train Network: Create and train a neural network with the specified training set and training parameters. Note: this button will enable when a training set and inputs are selected:

Classify Image: Classifies an image or group of images with the current neural network. Note: this button will only be enabled when a neural network has been trained or when an existing network has been loaded from file.

Save Network: Saves the currently trained neural network for future use. Note: this button will only be enabled when a neural network has been trained or when an existing network has been loaded from file.

Load Network: Loads a previously trained neural network from file.