Testing Saliency Parameters for Automatic Target Recognition

Sagar Pandya, Summer Internship Program
University of Southern California, Los Angeles, CA 90089

Mentor: Dr. Thomas T. Lu
Jet Propulsion Laboratory, Pasadena, CA 91109

ABSTRACT

A bottom-up visual attention model (the saliency model) is tested to enhance the performance of Automated Target Recognition (ATR). JPL has developed an ATR system that identifies regions of interest (ROI) using a trained OT-MACH filter, and then classifies potential targets as true- or false-positives using machine-learning techniques [2]. In this project, saliency is used as a pre-processing step to reduce the space for performing OT-MACH filtering. Saliency parameters, such as output level and orientation weight, are tuned to detect known target features. Preliminary results are promising and future work entails a rigorous and parameter-based search to gain maximum insight about this method.

1. Introduction

Automatic target recognition (ATR) is a focal area in computer vision and artificial intelligence research with wide applications in surveillance, navigation, medical imaging, and other fields. ATR is inherently difficult due to the complexity and variety of targets found in data sets. JPL has developed a multi-stage ATR pipeline that can be trained for different types of inputs (Figure 1, top). The first stage uses an OT-MACH filter to detect regions-of-interest (ROIs). In the next stage, features are extracted from these ROIs using principal component analysis (PCA). Finally, ROIs are classified as true- or false-positives by a trained machine-learning algorithm [3]. The aim of this summer project was to augment or replace the first stage of the ATR pipeline with a saliency model to detect preliminary ROIs (Figure 1, bottom). Two main datasets were used for testing: short range SONAR data, and visual data (Figure 2). Different saliency parameters were explored for the different datasets.

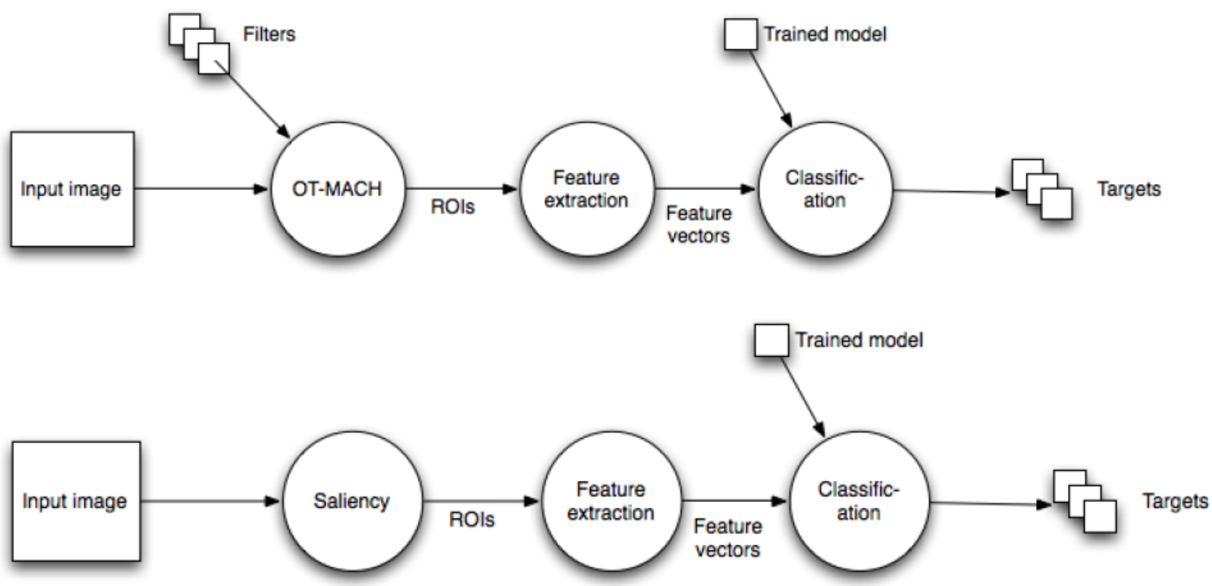

Figure 1. Top: Classic Multi-stage ATR System. Images are filtered by OT-MACH to find preliminary ROIs, which are then passed onto feature extraction and classification for final target discrimination. Bottom: saliency-ATR System, replacing the first stage of ATR with the saliency algorithm.

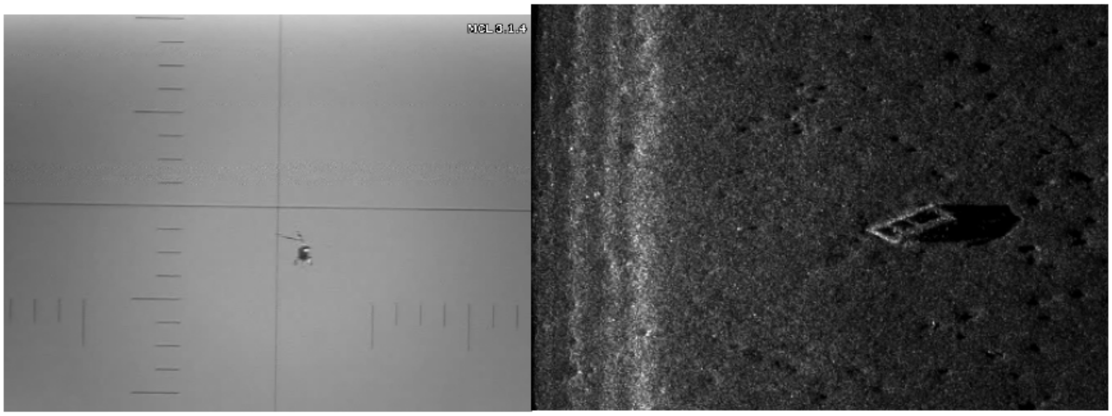

Figure 2. Samples of the types of input images used. Helicopter image (left) and SONAR image (right).

2. Background

This summer project was motivated by previous experience and work at the iLab at USC, advised by Dr. Laurent Itti. Much of iLab's work has been in the development of neurologically plausible visual models, one of which is the saliency model [1]. The saliency model effectively finds ROIs in natural scenes, and the main focus of this summer's work was to find out if saliency would be effective in the context of ATR. To be considered effective, saliency would have to perform as good as, or better than, the first stage of the ATR pipeline for finding ROIs.

2.1 ATR System

The ATR system developed at JPL breaks target recognition into three steps: preprocessing, feature extraction, and classification. This modular approach allows new methods to be tested independently without compromising the general ATR system requirements. For example, the preprocessing step can be any technique that produces ROIs before features are extracted from them. In the case of the general purpose ATR system, this preprocessing step is fulfilled by a trained OT-MACH filter (Figure 3).

A target-matching filter is constructed using a wavelet function, and then correlated with the input image in the Fourier domain [2]. This produces sharp peaks in the input image where likely targets can be found. These peaks are the ROIs. This first stage of the pipeline determines the best performance of the entire ATR system. If a target is not detected in this stage, then it will never be detected [3]. Therefore, it is important for the first stage to find every single possible target, even at the expense of including many false-positives. The results of the first stage are the ROIs, which are then passed onto the feature extraction and classification stages.

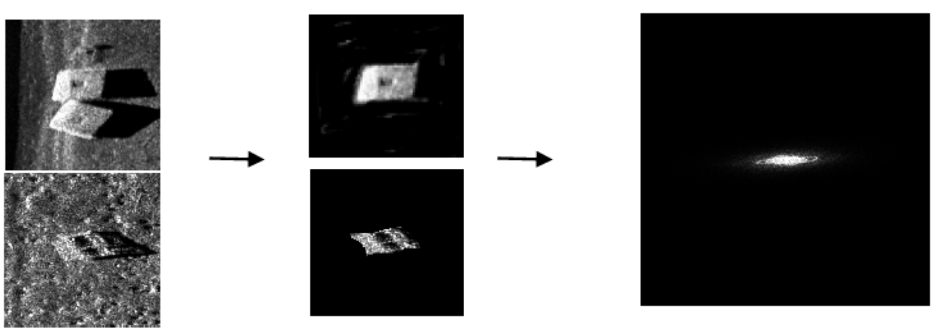

Figure 3: Examples of OT-MACH training set and filters on short range SONAR data. Training samples take account of different sizes and lighting conditions of targets. Left shows image samples of short-range data, middle shows training images, and right shows the produced filter in the spatial. [3]

The feature extraction and classification stages are responsible for stripping out false-positives from the first stage of the ATR system (Figure 4). The feature extraction stage parameterizes each ROI as a feature vector for classification. It has two important functions: to provide features that the classifier can use to differentiate between true- and false-positives; and to reduce the dimensionality of the feature space. Without this second function, each feature would be as large as the number of pixels in the ROI, which would greatly increase the computational time of this stage. Instead, the ATR system uses principal component analysis to reduce the feature space, and also find the best representative features of each ROI.

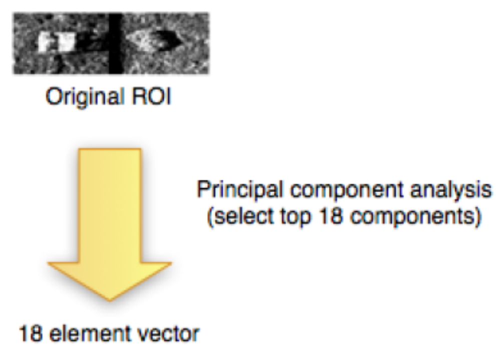

Figure 4

Finally, the classification stage takes each parameterized ROI feature vector and attempts to separate the true-positives from the false-positives. Any machine learning technique can be applied here, such as a back-propagation neural network or a support vector machine. In the ATR system used by this project, the Adaboost algorithm is used [3]. No matter what method is used, prior training must be performed for the classification to be successful.

2.2 Saliency Model

The saliency algorithm simulates bottom-up visual attention in primate visual cortexes. An input image is provided, and an output "saliency map" is produced that indicates ROIs. These ROIs are determined to be the most salient parts of the image, which a primate observer would be most likely to direct its attention to. This algorithm is fast and effective at finding visually conspicuous areas of an image. The algorithm works by breaking the input image into several feature maps, performing transformations on those maps, and then recombining them into a final saliency map [1] (Figure 5).

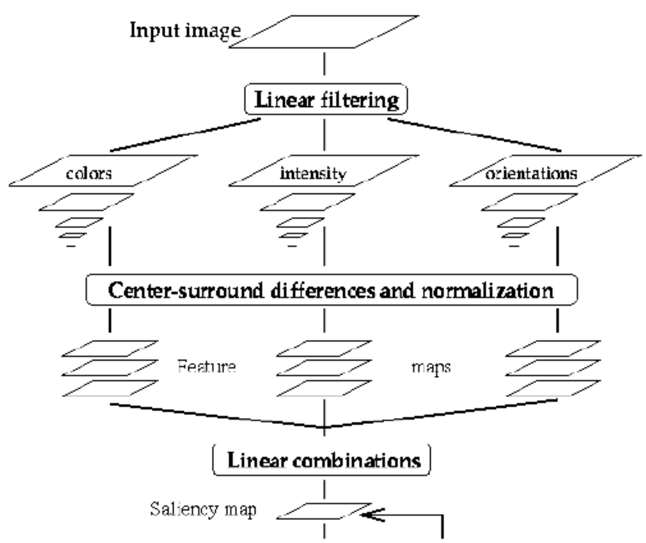

Figure 5: Saliency model [1]

Feature maps are extracted from the input image in three modalities: color, intensity, and orientation. Each feature map is down-sampled and scaled multiple times to produce an image pyramid. Four color maps are produced to represent red, green, blue, and yellow. These are computed as follows: $R = r - (g + b)/2$ for red, $G = g - (r + b)/2$ for green, $B = b - (r + g)/2$ for blue, and $Y = (r + g)/2 - |r - g|/2 - b$ for yellow. Intensity is computed as the average of the red, green, and blue channels. Orientation is computed from the intensity map using Gabor filtering. Across scales, a center-surround operation is performed, which uses surrounding information to inhibit feature responses. This simulates the behavior of neurons in the visual cortex, and is well suited to responding to locations that stand out from their surroundings [1]. The center-surround operation is implemented as cross-scale differences within the feature map image pyramids.

The resulting feature maps are combined at a specific pyramid level (the "output level") into a saliency map, which encodes information about salient regions of the input image. Depending on the application, these feature maps can be recombined in a weighted manner. These weights, along with the output level of the final saliency map, and the scale differences in the center-surround operation, encompass the parameters that saliency exposes for fine-tuning.

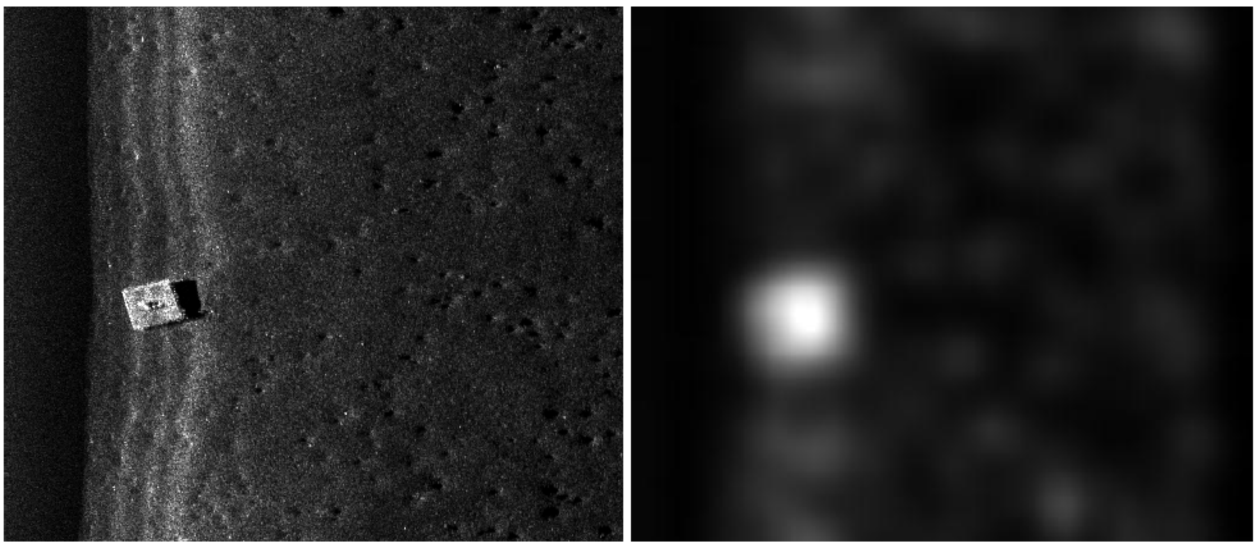

Figure 6: Example of saliency output (right) on short-range SONAR data (left). There is a strong peak where the target is, producing a very nice ROI for later ATR stages.

3. Objectives and Approach

The goals of this summer project were two-fold: to research new methods for ATR, and to increase the run-time performance of ATR. Along those lines, saliency was investigated as a way to accomplish both of these tasks. If saliency could be demonstrated to match or beat the OT-MACH filter's performance at finding ROIs, then it would also simultaneously increase the run-time performance of ATR since the saliency algorithm is computationally efficient. To test saliency's efficacy on the datasets, an implementation would be required that could allow for rapid testing with the existing ATR codebase. Therefore the development of a saliency-based ATR system would entail several steps:

- Test saliency on images from the ATR dataset for feasibility
- Port an existing implementation to Matlab and export relevant parameters as Matlab arguments
- Tune parameters per dataset
- Replace or augment the first ATR stage with a tuned-saliency model

Testing of saliency on our datasets showed promise for the algorithm, so work was begun to port the existing C implementation to Matlab. This involved restructuring the saliency model implementation into a so-called Mex-function that can be compiled into a shared object that Matlab can use. Once this was completed, the saliency model was fitted into the ATR system. Preliminary results demonstrated that saliency would need more fine-grained tuning per dataset before it could be considered useful for ATR, due to the sensitivity of output maps to input parameter choices (Figure 8).

4. Results

The saliency model parameters tuned for this application were weights for the color, intensity, and orientation channels; the output resolution; and the center-surround step sizes (Figure 7). The channel weights had the most negative effect, producing the best results when scaled almost all the way down. The output resolution parameter provided a tradeoff between noisy data, and low granularity. That is, the lower the output resolution, the less noisy the saliency map, but also the less detailed it could be. Thus for almost all cases, a mid- to high- output resolution was most effective. The center-surround step sizes effectively controlled the size of salient targets found, and how much they tended to stand out from their surroundings. These parameters were the most sensitive to input data, and required the most thorough tuning.

Due to the size of the parameter space, an automatic, unsupervised parameter-tuning algorithm is called for. This requires an appropriate scoring metric. The metric chosen was a ratio between the number of high peaks in the known target regions to the number of high peaks outside of the target regions. The optimal parameter set would maximize this ratio, producing very high peaks inside the target regions, and very low peaks elsewhere. This scoring metric proved to be robust for both datasets used in this summer project.

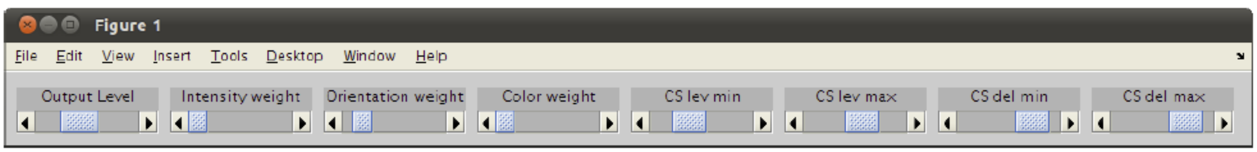

Figure 7: Parameter tuning interface showing good parameter selections for helicopter data.

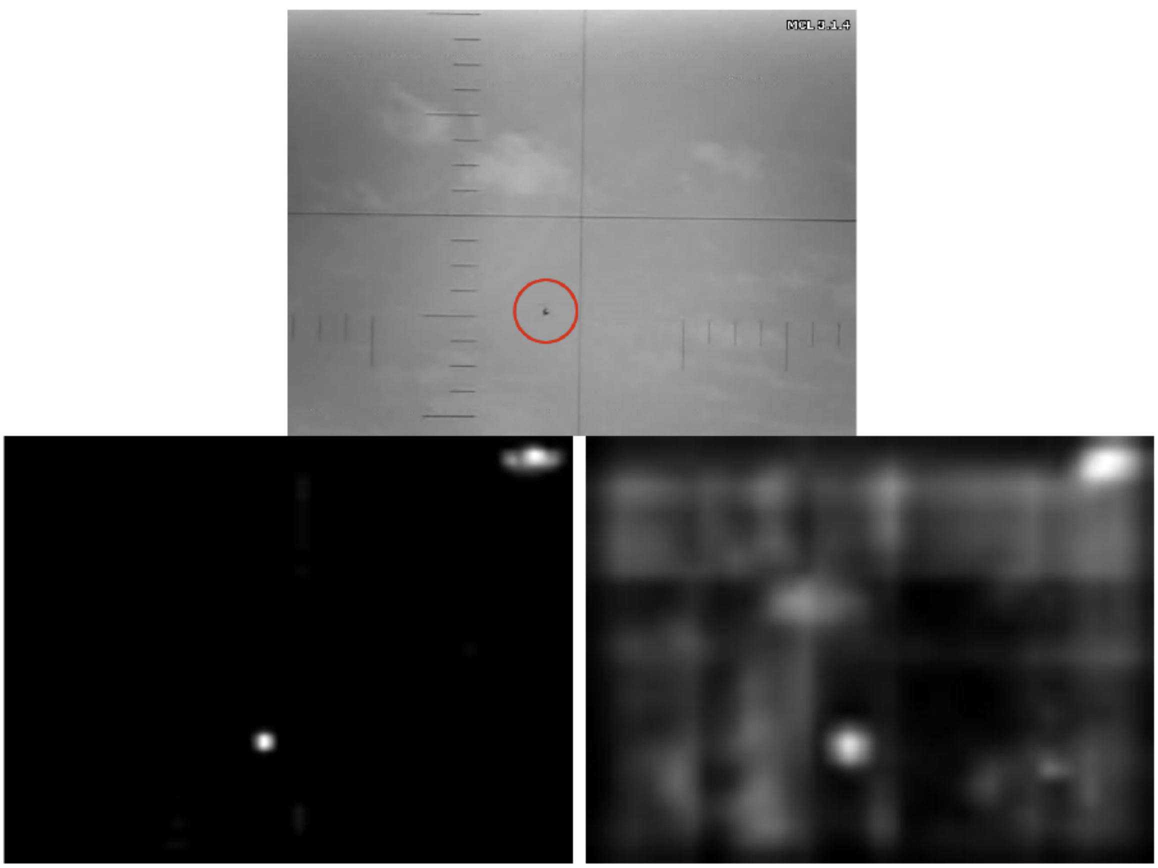

Figure 8: Comparison of good (bottom left) vs. bad (bottom right) saliency parameters on helicopter data (top). Helicopter appears in circled area in top image. Parameters used in good output shown in Figure 7.

5. Discussion

The results of this work support the intuition that saliency produces good results on datasets that are most like natural scenes. With well-tuned parameters and a good dataset, saliency outperforms OT-MACH. However, saliency's drawbacks prevent it from being a simple drop-in replacement for OT-MACH. These drawbacks include extensive parameter tuning per dataset, and preprocessing required on datasets where the targets are indistinguishable from background noise. These setup steps obviate one of the purposes of using saliency; namely, to avoid the filter-mv creation setup for OT-MACH.

Saliency does show promise for continued application to ATR. Future work should focus on a fast parameter search strategy. Hill climbing is suggested as one good search strategy, but future work would entail a full implementation and comparison of different search strategies. Integration of saliency feature map information into the feature extraction stage of ATR may also produce much better results. Combining these phases could improve the overall performance and execution time of ATR.

6. Acknowledgements

This work was performed at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the Summer Internship Program (SIP) and the National Aeronautics and Space Administration (NASA). Thanks to my mentor, Dr. Thomas Lu, for the guidance, support, and advice that made this project possible. Also thanks to Dr. Laurent Itti from the University of Southern California's iLab for granting me permission to use the saliency model and implementation developed in his lab; and Rand Voorhies, also from iLab, for his guidance in understanding the saliency model. Finally, thanks to Matthew Scholten and Brian Walker for their help familiarizing me with the ATR implementation at JPL.

7. References

1. Itti, L.; Koch, C.; Niebur, E.; , "A model of saliency-based visual attention for rapid scene analysis ," Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol.20, no.11, pp.1254-1259, Nov 1998

2. Johnson, O., Edens, W., Lu, T., and Chao, T., "Optimization of OT-MACH Filter Generation for Target Recognition," SPIE Conference 7340, (2009).

3. Lin, H., and Lu, T., “Optimization of Adaboost Algorithm for Sonar Target Detection in a Multi-Stage ATR System,” JPL USRP Final Report, (2010).