REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
10-06-2015	Final Report	15-Aug-2014 - 14-May-2015

4. TITLE AND SUBTITLE
Final Report: Activity Detection and Retrieval for Image and Video Data with Limited Training

5a. CONTRACT NUMBER
W911NF-14-1-0385

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Prof. Scott T. Acton, Prof. Zongli Lin, Rituparna Sarkar

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
University of Virginia
Office of Sponsored Programs
P. O. Box 400195
Charlottesville, VA 22904 -4195

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
66186-CS-II.3

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
The main objective of this project is to exploit image analysis tools for solving image segmentation and classification applications. Here we propose two techniques for image segmentation. The first involves an automata based multiple threshold selection scheme, where a mixture of Gaussian is fitted to the image intensity histogram, and the parameters for each of the Gaussian distribution are estimated by learning automata. For our second approach to segmentation, we employ a region based segmentation technique that is capable of handling intensity inhomogeneity, and use a sparse representation based dictionary learning technique to learn the basis function from

15. SUBJECT TERMS
image processing, image analysis

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Scott Acton
UU	UU	UU	UU		19b. TELEPHONE NUMBER 434-982-2003

Report Title

Final Report: Activity Detection and Retrieval for Image and Video Data with Limited Training

ABSTRACT

The main objective of this project is to exploit image analysis tools for solving image segmentation and classification applications. Here we propose two techniques for image segmentation. The first involves an automata based multiple threshold selection scheme, where a mixture of Gaussian is fitted to the image intensity histogram, and the parameters for each of the Gaussian distribution are estimated by learning automata. For our second approach to segmentation, we employ a region based segmentation technique that is capable of handling intensity inhomogeneity, and use a sparse representation based dictionary learning technique to learn the basis function from a given set of training data. The image intensities of the test image to be segmented are then represented as a linear combination of these learned bases. We also implement a kernel based dictionary learning scheme for image classification. Here we extract multiple feature types from the images and learn a dictionary based on the combination of their kernel representation, and implement a mutual information based technique for determining the kernel combination weights. The final classification is based on the minimum reconstruction error for these features. We provide detailed description and supporting experimental results for each of these methods.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations

Number of Presentations: 0.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received	Paper

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received	Paper

TOTAL:

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

Received		Paper
06/10/2015	1.00	Qian Sang, Zongli Lin, Scott T. Acton. Learning Automata for Image Segmentation, Pattern Recognition Letters (05 2015)
06/10/2015	2.00	Rituparna Sarkar, Suvadip Mukherjee, Scott T. Acton. Dictionary Learning Level Set, IEEE Signal Processing Letters (04 2015)

TOTAL: **2**

Number of Manuscripts:

Books

Received Book

TOTAL:

Received Book Chapter

TOTAL:

Patents Submitted

Patents Awarded

Awards

Scott Acton became Editor-in-Chief, IEEE Transactions on Image Processing
Scott Acton was named IEEE Fellow

Graduate Students

NAME	PERCENT_SUPPORTED	Discipline
Rituparna Sarkar	0.50	
FTE Equivalent:	**0.50**	
Total Number:	**1**	

Names of Post Doctorates

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Names of Faculty Supported

NAME	PERCENT_SUPPORTED	National Academy Member
Scott Acton	0.04	
Zongli Lin	0.04	
FTE Equivalent:	**0.08**	
Total Number:	**2**	

Names of Under Graduate students supported

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: 0.00

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:..... 0.00

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):..... 0.00

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense 0.00

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields:..... 0.00

Names of Personnel receiving masters degrees

NAME
Total Number:

Names of personnel receiving PHDs

NAME
Total Number:

Names of other research staff

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Sub Contractors (DD882)

Inventions (DD882)

Scientific Progress

See attached.

Technology Transfer

Final Report

Report for ARO STIR Project
(Contract No. 146380-101-GG11756-31335)

PI and co-PIs

Prof. Scott Acton & Prof. Zongli Lin

Project team

Prof. Scott T. Acton,

Prof. Zongli Lin,

Rituparna Sarkar

University of Virginia, Charlottesville, USA

Table of Contents

List of Figures

List of Tables

1. Introduction

The goal of this work is to exploit image analysis techniques to solve image segmentation and image retrieval problems. Here we propose two segmentation techniques; the first method involves an automated threshold selection method, while the second is a region based segmentation method where the intensity inhomogeneity is modeled as a linear combination of the basis learned from the images present in the dataset. The image classification method implemented here is based on the multiple-kernel based dictionary learning technique.

A wide variety of threshold selection techniques for image segmentation exist in the literature. Conventional methods pick a threshold that minimizes the overlap or maximizes the variance in between two adjacent intensity histogram clusters, based on certain statistical metric. We propose an alternative threshold selection in the context of a GMM where the threshold selection becomes a two-step process: first, estimate the Gaussian distribution parameters (mean, standard deviation, and weight) by fitting the mixture of Gaussian that best approximates the normalized histogram; and then, calculate thresholds based on the estimated Gaussian parameters. Here, we employ a learning-based optimization approach using learning automata for estimating Gaussian mixture parameters. The randomness in the generation of parameter estimates provides learning automaton algorithms with obvious advantages over the gradient-based and expectation maximization (EM) algorithms. We approach the Gaussian parameter estimation problem with the so-called continuous action reinforcement learning automata (CARLA) [**1**] algorithm. In particular, the output of each automaton corresponds to one specific Gaussian parameter to be estimated. Their combination at each stage provides a fitted Gaussian mixture model of the original image histogram. The match between the two is evaluated based on some metric, e.g., the average mean square error, and is returned as a reinforcement signal, to be applied as the input to each automaton. Inside a learning automaton, a probability distribution function (PDF) is defined over the parameter search range, representing the desirability of each specific value, and is updated as learning proceeds based on the reinforcement signal. A parameter estimate an automaton output) is randomly generated for the next stage based on the current probability distribution.

The second approach for segmentation employs a novel region based segmentation technique using dictionary learning. We hypothesize that in problems where a set of training images for the object is available for analysis, each image can be compactly represented as a linear combination of similar images or a basis learned from the training images and segmentation accuracy can be significantly improved by computing the basis functions instead of specifying them implicitly. The salient idea of this approach is to compute the optimal set of functions to model the region intensities. Our solution to this problem involves the integration of a level set segmentation methodology with the dictionary learning framework. This provides a solution to deal with intensity inhomogeneity prevalent in many imaging applications such as ultrasound and fluorescence microscopy. The primary objective of this approach is to develop a region based segmentation framework, which is capable of handling intensity inhomogeneity. It may be debated that adding edge information to the region based framework can improve segmentation. However, extracting accurate edge map is a challenging issue by itself for applications where the signal is poor due to presence of speckle and clutter. Therefore, it is of considerable interest to develop a solely region dependent technique that can accommodate artifacts such as noise, clutter and illumination variation. This approach employs the dictionary learning technique to obtain the basis function (learn from the training data) that approximates the image region intensities. This can be viewed from the perspective of low dimensional approximation of a

signal. While Chan-Vese's [**2**] method is a form of extreme dimensionality reduction (due to piecewise constant assumption), out method achieves a balance between reduction of dimensionality and accurate intensity modeling.

The third approach deals with a method for image classification problem. Significant development in sparse representation based classification methods [**3**] [**4**] [**5**] [**6**] [**7**] has come about in the past few years. In [**4**], it was shown that a test image can be represented as a linear combination of only a few of the training images and is best represented when the test image is a linear combination of images belonging to the same class. However, it has been shown in [**5**] [**7**] [**6**] [**3**], instead of using a pre-specified basis for representation of the test sample, learning the bases from the training data itself has proved to be more effective in data representation. But the dictionary learning presented in these papers is based on a linear model in feature space. The main problem with this approach is that they cannot capture the non-linearity present in the data. Kernel learning methods are widely used to deal with non-linearity in the feature space. A non-linear transform is performed on the feature space to convert the data to a higher dimensional subspace and then a linear classifier model can be applied. In a classification system the choice of feature extracted from the image is of crucial importance. However, a single feature cannot discriminatively represent all the images in the dataset. Here, we develop an information theoretic kernel combination method embedded in the dictionary learning framework. One advantage of the kernel space representation, other than a higher-dimensional representation, lies in the fact that different features can be combined in the kernel space using multiple kernel functions. Our method learns a dictionary in the kernel space for each of the classes in the learning phase. We employ a mutual information based approach to obtain the most desirable weights for kernel combination. In the testing phase, our method exploits the learned dictionaries and the kernel weights to assign a class label to the test image.

In this report we provide a background for each of the methods implemented in solving the above mentioned image analysis tasks. Then we describe in details the methodologies for the threshold selection, region based segmentation and the image classification problem. We provide relevant experimental results for each of these approaches and discuss the future directions for these applications.

2. Background

2.1. Level set Segmentation

The level set method, used as a numerical technique to track shapes can be applied to image segmentation techniques. Image segmentation solely based on the gray values of image pixels was first proposed by Mumford and Shah [**8**]. This region based approach was made popular later by Chan and Vese [**9**] who used level sets to propagate geometric snakes to segment the image into constant intensity regions. The Chan-Vese framework proposes to partition the image $\mathrm{f}(\mathbf{x})(\mathrm{x} \in \Omega \subseteq \mathbb{R}^2)$ into sets of constant intensity regions. The optimal partition is obtained by trying to minimize the following energy functional:

$$\int_{\Omega} |\mathrm{f(x)} - \mathrm{c}_1|^2 \mathrm{m}_1(\mathrm{x})\, \mathrm{dx} + \int_{\Omega} |\mathrm{f(x)} - \mathrm{c}_2|^2 \mathrm{m}_2(\mathrm{x})\, \mathrm{dx} \qquad (1)$$

Here ϕ is a level set function whose zero level set denotes the object boundary. ϕ is constructed such that its value is positive inside the zero level contour and negative outside. The local minimizer ϕ^* of (1) partitions the image such that the two region intensities are best approximated by the constant scalars c_1 and c_2, which are updated iteratively using alternating

minimization. $m_1(\boldsymbol{x}) = H_\epsilon(\phi)$ is the regularized version of the standard Heaviside function, the extent of regularization being controlled by the parameter $\epsilon, m_2(\boldsymbol{x}) = 1 - m_1(\boldsymbol{x})$.
Often in practical scenarios a constant intensity model fails to capture the intensity inhomogeneity. To adapt the model in (1) to capture heterogeneous intensity regions, the scalar c_1, c_2 are replaced with smooth functions $c_1(\mathbf{x})$ and $c_2(\mathbf{x})$ respectively and the smoothness is enforced by constraining the total variations of these functions.

2.2. Dictionary Learning

Sparse coding can be efficiently utilized by representing an image (or a signal) as a linear combination of some basis vectors. This can be written as $\mathrm{F} = D\mathrm{Y}$, where F is an image (or a signal), D is a matrix in which columns represent the basis vectors, which we call dictionary, and Y contains the representative sparse codes. Given a set of training data, the goal of dictionary learning [**10**] [**11**] is to compute a set of basis vectors, also called *atoms*, such that each training data can be represented as a linear combination of only a few of these atoms. The key idea is to utilize the underlying sparsity of the training data, while minimizing the reconstruction error. If $\boldsymbol{F} = [\mathrm{f}_1, \dots, \mathrm{f}_N]$ denotes the set of N training images, we can use dictionary learning technique to compute the dictionary $\boldsymbol{D}_k = [d_1, \dots, d_k]^T$ by solving the following optimization problem

$$\mathrm{D_k} = \underset{\mathrm{D},\mathrm{y_i}}{\operatorname{argmin}} \left|\left|\mathrm{f_i} - \mathrm{D^T y_i}\right|\right|_2^2$$
$$\text{such that } \left|\left|\mathrm{y_i}\right|\right|_0 \leq \theta, \qquad \forall\, \mathrm{i} = 1, \dots, \mathrm{N}. \tag{2}$$

where y_i is a coefficient vector corresponding to the i^{th} training image and θ is a scalar which dictates the level of sparsity. There are a number of methods in the literature that use some approximation to solve the hard optimization problem. Dictionary learning exploits sparsity in the data (2) by constraining l_0 norm of the coefficients.

2.3. Multiple Kernel Learning

To deal with non-linearity in the data, the kernel approach is applied which non-linearly transforms a data to a higher dimensional space. As proposed in [**12**] [**13**] [**14**] [**15**], linear learning methods can be generalized to non-linear learning methods by this non-linear transformation. Let ϕ: $\mathbb{R}^N \rightarrow \mathcal{H}$, be the non-linear transformation that transforms the features to a higher dimensional kernel space referred to as reproducing kernel Hilbert space (RKHS). In this space, the data $Y = [\boldsymbol{y}_1, \boldsymbol{y}_2, \dots\dots, \boldsymbol{y}_{N_c}]$ can now be represented as $\boldsymbol{\phi}(Y) = [\boldsymbol{\phi}(\boldsymbol{y}_1), \boldsymbol{\phi}(\boldsymbol{y}_2), \dots\dots, \boldsymbol{\phi}(\boldsymbol{y}_{N_c})]$. A kernel function $K(.)$, yields the inner product of any given two vectors in RKHS such that $K(y_i, y_i) = \phi(y_i)^T \phi(y_i)$. The kernel function $K(.)$ gives a measure of the similarity in a reproducing kernel Hilbert space and in RKHS can be represented as a linear combination of other valid kernels where the linear coefficients are non-negative such that:

$$\mathcal{K} = \sum_{s=1}^{\mathcal{S}} \beta_\mathrm{s} \mathrm{K_s} \tag{3}$$

where $\mathcal{K}$ is the kernel function obtained from a linear combination of $\mathcal{S}$different kernel functions (for $\mathcal{S}$different features). We define the linear combination of the kernel matrix $\mathcal{K} = \sum_{s=1}^{\mathcal{S}} \beta_\mathrm{s} \mathrm{K_s}(.)$, such that $\Psi(.)^\mathrm{T}\Psi(.) = \mathcal{K}$. Ψ is defined as the non-linear transform from feature space to RKHS.

2.4. Mutual Information:

Mutual information between two random variables provides a measure of dependence on one another. Higher the mutual information greater is the dependency. A relevance measure between features and the class they belong to can be obtained by maximizing the mutual information [**16**] [**17**] [**18**] [**19**]. For a given feature $\boldsymbol{y}$ the mutual information between the feature and its class $\ell(\boldsymbol{y}) = i$ is given by (15).

$$\mathbb{I}(\boldsymbol{y}, \ell(\boldsymbol{y}) = i) = H(i) - H(i|\boldsymbol{y}) \tag{4}$$

where $H(i)$ is the entropy given by:

$$H(\boldsymbol{y}) = p(\boldsymbol{y}) \log\left(\frac{1}{p(\boldsymbol{y})}\right) \tag{5}$$

For a given class, the class entropy $H(c)$is constant. Thus maximizing the mutual information between a feature and a class would mean minimizing the conditional entropy $H(c|\boldsymbol{x})$, where the conditional entropy is given by

$$H(c|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|c)p(c)}{p(\boldsymbol{x})} \log\left(\frac{p(\boldsymbol{x})}{p(\boldsymbol{x}|c)p(c)}\right) \tag{6}$$

In the above equation, $p(\boldsymbol{x}) = \sum_{c=1}^{C} p(\boldsymbol{x}|c)p(c)$.

The mutual information provides a measure of a feature belonging to a class. Thus minimizing the conditional entropy ensures a more compact distribution of the data belonging to the same class or in other words increases the discrimination between class distributions [**20**]. If f denotes the function transforming the given feature vector to the RKHS, we can re-write the entropy for the transformed feature as $H(c|f(\boldsymbol{x}))$.

2.5. Image features

Scale-invariant feature transform (SIFT): The SIFT algorithm looks for important pixels to summarize the entire image. Once the algorithm determines the important pixels (i.e., the descriptors), then it computes local gradient histograms around each descriptor to form a 128 dimensional features for each descriptor. More details can be found in [**21**].

Histograms of oriented gradients (HOG): The HOG algorithm forms small overlapping image patches from a given image and then for each patch, it computes local gradient based histograms. More details on HOG can be found in [**22**].

2.6. Learning automata

Inside a learning automaton, a probability distribution function (PDF) is defined over a parameter search range, representing the desirability of each specific value and is updated as learning proceeds based on reinforcement signal. A parameter estimate (an automaton output) is randomly generated for the next stage based on the current probability distribution.

A typical work flow of a learning automaton is shown in Figure 1: **Basic learning procedure of learning automata**, and is described as follows:

1. At time *n*, the automaton generates an action $\alpha(n)$ from the current probability distribution $P(\alpha_i), i \in \{1,2,\dots,r\}$, over the action set, i.e., $\alpha(n) \in \{\alpha_1, \alpha_2, \dots, \alpha_r\}$, where $P(\alpha_i)$ denotes the probability that α_i is picked as the current action to be applied to the environment. Note that $\sum_{i=1}^{r} P(\alpha_i) = 1$.
2. Action set is $\alpha(n)$ applied to the environment.
3. The environment reacts to the action $\alpha(n)$, and its performance is evaluated to produce a response $\beta(n) \in \{\beta_1, \beta_2 \dots \beta_m\}$.

4. Based on this response, $P(\alpha_i)$ is updated for $i \in \{1,2,\dots,r\}$.
5. Steps 1–4 are repeated until the process converges or some stopping criterion is satisfied

Different probability updating procedures distinguish a variety of learning automata algorithms. Their performance and convergence properties have been documented in [**23**].

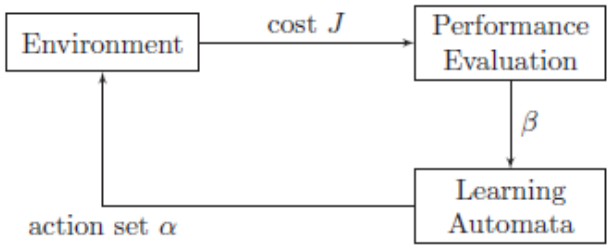


Figure 1: Basic learning procedure of learning automata

Continuous action reinforcement learning automata (CARLA): This learning automaton proposed in [**1**] extends the discrete structure of a typical learning automaton to the case when the action set, as well as the response from the environment, is characterized by a continuous variable. Accordingly, the discrete probability distributions of the action set are replaced by PDFs over a domain, on which a potential action is defined. It is used as a function optimizer, and fits in the image segmentation problem under consideration well. The action by the automaton corresponds to a parameter used by the specific segmentation algorithm, i.e., the Gaussian parameters. For each segmentation parameter, there is a corresponding automaton. The environment is the input image histogram to be processed by the algorithm.

The response is a certain criterion (e.g., root mean square error between the image histogram and its model as a fitted Gaussian mixture) on the segmentation performance with the current segmentation parameters generated from the *PDF*s of the automata. CARLA is a linear reward/inaction learning scheme [**23**]. The basic steps of CARLA is as follows:

1. With a current response $\beta(n)$ at the n^{th} iteration, the PDF $p(x;n+1)$ of the parameter to be estimated is updated by adding a Gaussian $G(x;\alpha(n))$ centered at $\alpha(n)$ with a certain height λ and standard deviation σ, i.e.,

$$G\big(x;\alpha(n)\big) = \lambda \exp\left[-\frac{\big(x-\alpha(n)\big)^2}{2\sigma^2}\right], \qquad x \in \{x_{min}, x_{max}\} \tag{7}$$

where (x_{min}, x_{max}) designates the search range of the parameter under consideration. σ and λ are tuned with two free parameters g_w and g_h based on this range according to

$$\sigma = g_w(x_{max} - x_{min}), \text{and } \lambda = \frac{g_h}{x_{max}-x_{min}} \tag{8}$$

2. The PDF is then updated as follows for $x \in (x_{min}, x_{max})$

$$p(x;n+1) = c_s(n+1)\big[p(x,n) + \beta(n)G\big(x,\alpha(n)\big)\big] \tag{9}$$

Where, $c_s(n+1)$ is a scaling constant to ensure $p(x,n+1)$ is a PDF.

3. Suppose a cost or performance index of $J(n)$ is returned. Current and previous costs are stored in a reference set $R(n)$. The median and minimum values J_{med} and J_{min} are calculated, and $\beta(n)$ is defined as

$$\beta(n) = \max\left\{0; \frac{J_{med} - J(n)}{J_{med} - J_{min}}\right\} \tag{10}$$

4. The parameter estimate $\alpha(n+1)$ at the next time instant is randomly picked according to the updated PDF $p(x;n+1)$. In particular, a random number $r(n+1)$ within [0,1] is generated from a uniform distribution, and $\alpha(n+1)$ is picked such that

$$\int_{x_{min}}^{\alpha(n+1)} p(x, n+1)\, dx = r(n+1)$$

5. The learning procedure continues until a prescribed criterion or a stopping condition is met. The parameter estimate is then taken as the value corresponding to the maximum of the PDF. With the Gaussian parameters obtained from the learning process, thresholds can be computed for a minimum misclassification error or simply as the average of two adjacent Gaussian mean estimates.

3. Methods and experimental results

3.1. Image Segmentation

3.1.1. Learning automata for image segmentation

We consider the multi-level thresholding problem for an image [**24**] that has a normalized histogram that is modeled as a mixture of Gaussians. Each Gaussian component in this case is assumed to correspond to an object (or a group of objects with similar pixel grayscale level distributions) of interest. Two key issues to be addressed in this case:

a. the number of components in the GMM needs to be determined,
b. parameter estimates of each Gaussian component are required for threshold computations.

We adopt the method used in [**25**], where a Gaussian component is detected with a pair of zero crossings in the second order difference of the GMM. In an ideal Gaussian density distribution with a mean μ and a standard deviation σ, this pair of zero crossings occurs at the inflection points of the Gaussian, i.e., $\mu \pm\sigma$. Since the tails of the Gaussian components overlap with each other to a large extent in general, the locations of zero crossings cannot be used to precisely extract the individual component. We adopt an iterative learning procedure, in which locations of the zero crossings are used to initialize the search range for a given parameter, and a learning automaton [**26**] is employed to progressively refine the estimate. The Gaussian component detection and localization procedure outlined shown in Fig.Figure 2: **Gaussian component detection and parameter estimation with learning automata**Detailed description is provided as follows.

a. Gaussian Component Detection

The histogram of an image provides statistical information on its pixels with different grayscale levels. It is first normalized to be viewed as a probability density function $h(x), x \in [0;\ L-1]$, where L is the number of different grayscale levels used in the image representation. The normalized histogram $h(x)$ is then modeled as a Gaussian mixture $G(x)$. To avoid over segmentation histogram for a real image is first smoothed by a Gaussian kernel with a chosen standard deviation τ.

$$G_\tau(x) = \frac{1}{\sqrt{2\pi\tau}} \exp\left(-\frac{x^2}{2\tau^2}\right) \tag{11}$$

This gives a smoothed normalized histogram $h_\tau(x) = G_\tau(x) * h(x)$, '*' denotes the convolution operation.

Zero crossing of the derivative approximation signal for $h_\tau(x)$ [**27**] can then be detected. Note that the zero crossings should appear in pairs with the sign of the second order difference of $h_\tau(x)$ changing from positive to negative at the left crossing point, and from negative to positive

at the right crossing point. These inflection points occur at $\mu \pm \sigma$ for an ideal Gaussian $G(\mu, \sigma)$, and provide initial estimates of its mean and standard deviation. Each pair of inflection points determines the presence of a Gaussian component. Suppose there are K pairs of inflection points. The Gaussian mixture model of $h_\tau(x)$ is expressed as:

$$G(x) = \sum_{i=1}^{K} \frac{p_i}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x-\mu)^2}{2\sigma_i^2}\right) \tag{12}$$

where, $\mu_i, \sigma_i \; and \; p_i$ are the mean, the standard deviation and the weight of the i^{th} component in the mixture respectively and $\sum_{i=1}^{K} p_i = 1$.

b. Gaussian Parameter Estimation with Learning Automata

A learning automaton can be considered as an optimizer that attempts to determine an optimal action (out of a set of potential actions $\alpha_1, \alpha_2, \ldots, \alpha_r$) to be applied to the external environment for the best performance (maximum benefit, minimum cost, etc.), based on a progressive learning process from the response $(\beta_1, \beta_2, \ldots, \beta_m)$ of the environment, in a stochastic way [**23**] as described in section 2.7. The desirability of the action set is represented by probabilities, which are updated as learning proceeds.

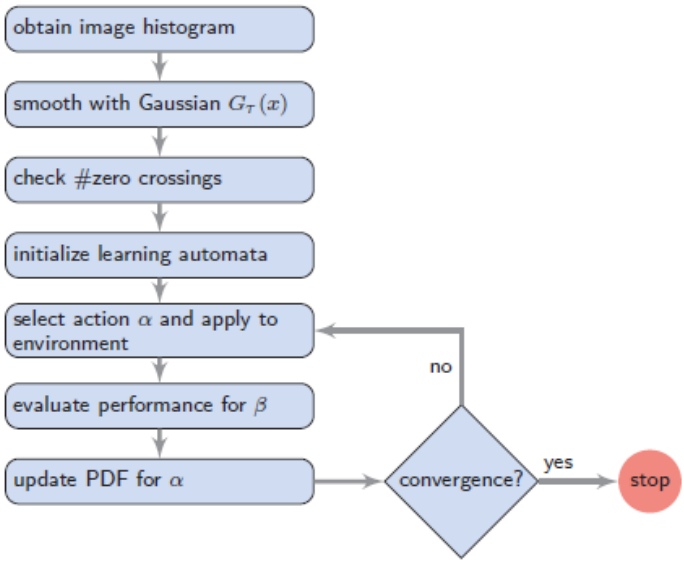


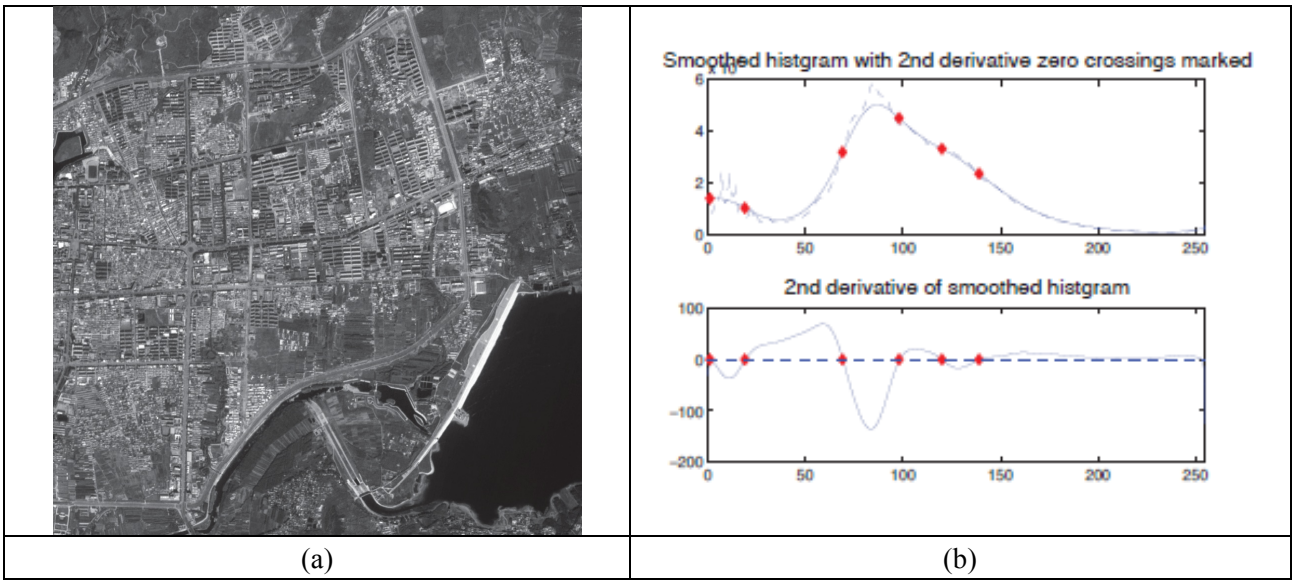
Figure 2: Gaussian component detection and parameter estimation with learning automata

Experimental Results

An example of the segmentation problem of a remotely sensed image Figure 3: (a) **shows an example of a remote-sensing image for segmentation and (b) shows the Gaussian component detection** is provided next to illustrate the descriptions and basic operations of the proposed thresholding method. Here we are interested in automatically computing suitable thresholds that can divide the image into three regions: man-made structures, the green area, and the water area. In this case, the image is assumed to contain three components of different grayscale level distributions, and two thresholds are sought. The Gaussian component detection procedure determines three pairs of zero crossings, i.e., K=3, as shown in Figure 3: (a**) shows an example of a remote-sensing image for segmentation and (b) shows the Gaussian component detection**. Therefore, there are eight Gaussian parameters to be estimated. Figure 4: **(a), (b) and (c) shows the three different Gaussian components and the segmentation results corresponding to that Gaussian component** shows the parameter learning process (evolution of the PDF) and the final segmentation results for each Gaussian component. For this example, the two threshold levels are simply chosen based on the Gaussian mean estimates with

$$T_i = \frac{\widehat{\mu_i} + \widehat{\mu_{i+1}}}{2}, i = 1,2, \ldots, K-1 \tag{13}$$

where $\widehat{\mu_i}$ is the estimate of μ_i , the mean of the i^{th} Gaussian component. The thresholds T_i can also be computed based on other criteria, e.g. minimum misclassification error [**28**], using the estimates of the standard deviations of each Gaussian component.



(a) (b)

Figure 3: (a) shows an example of a remote-sensing image for segmentation and (b) shows the Gaussian component detection

3.1.2. Image Segmentation using dictionary learning

Here we develop an *automated* segmentation algorithm, where a set of basis vectors is obtained from an existing dataset of images. . The main objective is that instead of explicitly specifying the set of basis elements; we estimate an optimal set of bases from the set of training images using dictionary learning. Similar images to be segmented are expressed as a linear combination of these basis vectors. As mentioned earlier, the constant intensity model fails to capture the intensity heterogeneity. Hence the basic Chan-Vese model [**2**], is generalized to make the model more flexible. In order to do so, we add higher order terms which can capture the intensity variations in the regions. Going by the intuition of Chan and Vese, it is fair to approximate the mean image of a dataset as a piecewise constant image. The two steps that comprises of the segmentation method are as follows:

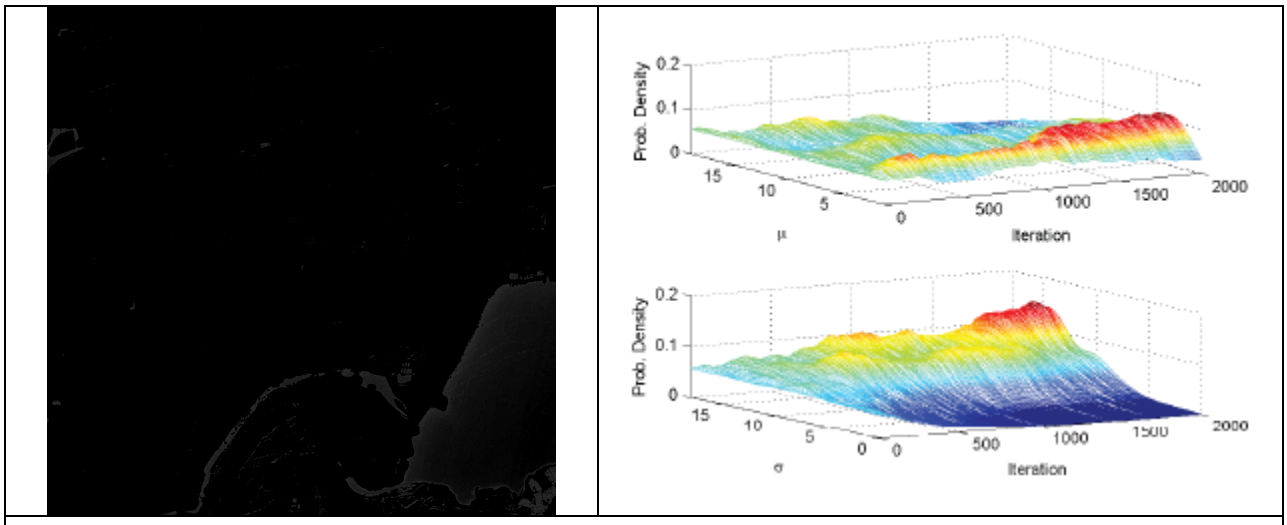
a. Dictionary learning

The first step involves obtaining the basis elements for a given dataset which gives a compact representation of all the images corresponding to that dataset. We employ K-SVD based [**21**] the dictionary learning discusses in section 2.2. We denote $\boldsymbol{D}_k(\mathbf{x}) = [d_1(\mathbf{x}), \dots, d_k(\mathbf{x})]^T$ as the dictionary learned from the mean subtracted images using (2).

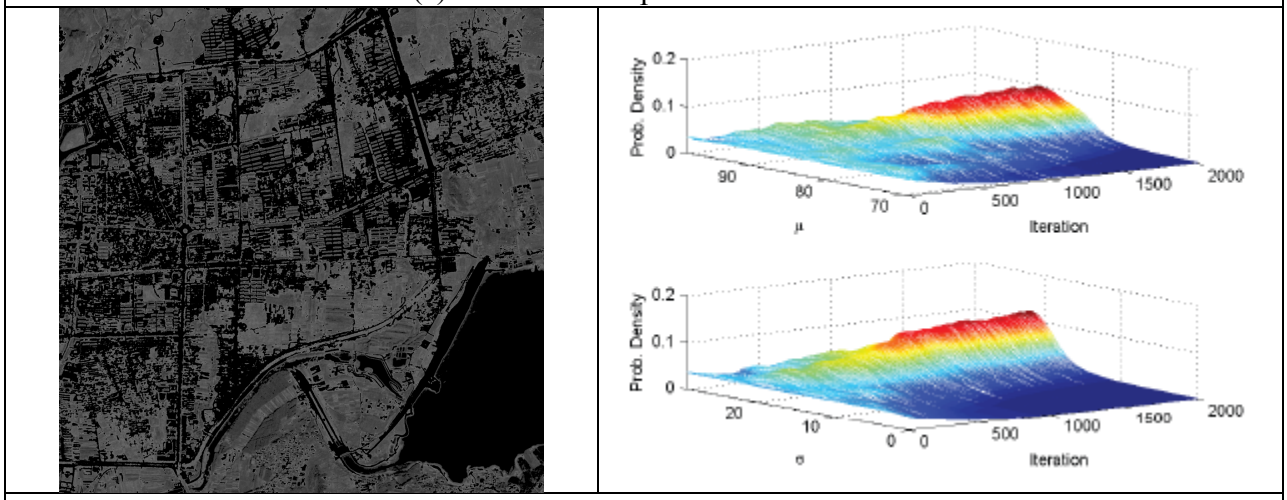
b. Level set segmentation using learned dictionary

Assuming a mean image which is approximately piecewise constant, the dictionary atoms learned from the mean subtracted dataset can be utilized to provide the non-linear variation necessary to model the intensity inhomogeneity [**29**]. A generalized version of Chan-Vese's model can be formulated as follows:

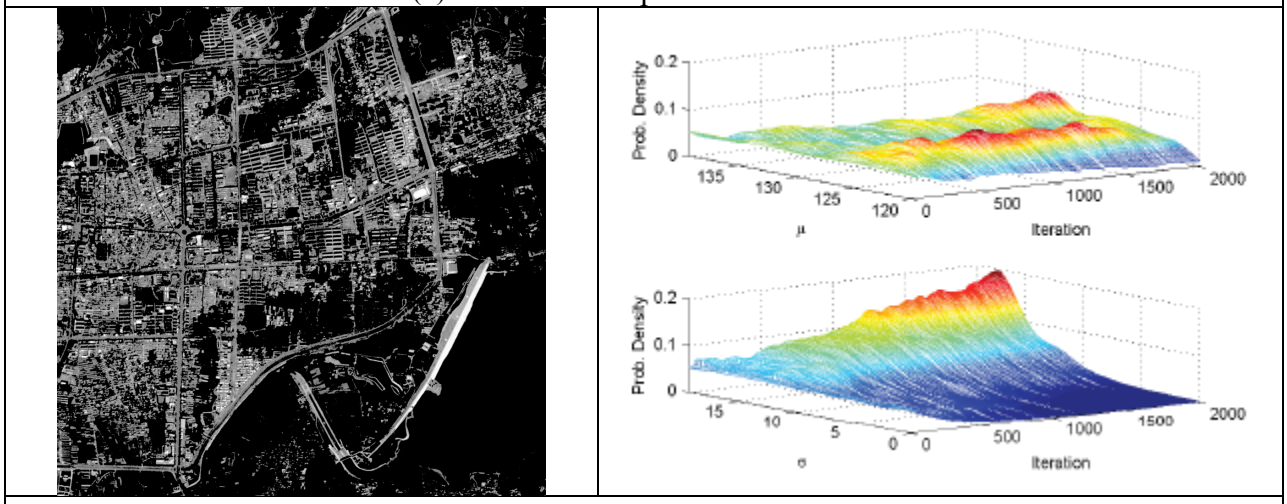
$$\mathcal{E}(\phi, A, B) = \int_{\Omega} \left| f(\mathbf{x}) - \sum_{i=0}^{k} a_i d_i(\mathbf{x}) \right|^2 m_1(\mathbf{x})\, d\mathbf{x} + \int_{\Omega} \left| f(\mathbf{x}) - \sum_{i=0}^{k} b_i d_i(\mathbf{x}) \right|^2 m_2(\mathbf{x}) d\mathbf{x} + \nu \int_{\Omega} |\nabla H_{\epsilon}(\phi)| d\,\mathbf{x} + \lambda \left(\left| |A| \right|_2^2 + \left| |B| \right|_2^2 \right) \tag{14}$$



(a) Gaussian component #1: Water area



(b) Gaussian component #2: Green area



(c) Gaussian component #3: Manmade structures

Figure 4: (a), (b) and (c) shows the three different Gaussian components and the segmentation results corresponding to that Gaussian component

Here $d_0(\mathbf{x}) = \mathbf{1}$ and $d_1, \ldots, d_k$ are dictionary elements or atoms which are used to model the non-linearity in the intra-region intensities of the images. The third term in (14) introduces smoothness in the solution, which is controlled using the parameter ν. $A = [a_0, \ldots, a_k]^T, B = [b_0, \ldots, b_k]^T$ are $(k+1)$ dimension real valued coefficient vectors. The parameter λ reduces over-fitting, by constraining the l_2 norm of the coefficient vectors.

In other words, (14) generalizes the traditional Chan-Vese technique by introducing capability to handle heterogeneous image regions. Here $d_1, \ldots, d_k$ can be interpreted as `detail functions' to model the intensity variation in conjunction to the constant illumination term d_0. Assuming a mean image which is approximately piecewise constant, the dictionary atoms are learned from the mean subtracted dataset and can be utilized to provide the non-linear variation necessary to model the intensity inhomogeneity. One can also think of the dictionary atoms as incorporating higher order details, learned to suit the dataset

Experimental results

We use five different sets of images to evaluate the performance of our algorithm. Out of them, three datasets contain images of medical phantoms, which mimic human veins. The remaining two datasets consists of human vein images, captured in vivo. Each dataset contains approximately 18 to 60 images, captured in C-mode using a portable, battery operated ultrasound scanner. The different images in a given set correspond to the image of a vein at various depths. Note that each dataset consists of registered blood vessel images. The vessel orientation and scale are also consistent. A separate dictionary is computed using the mean-subtracted images for each of the datasets.

Dependency on contour initialization: We show the performance of our algorithm using both manual and automatic initialization methods. The segmentation results with manual and automatic initialization for DL2S [**29**] Chan-Vese [**9**] and L2S [**30**] are shown in Figure 5: Comparison of segmentation results using manual and automatic initialization methods. (a) initialized contour (b) segmentation results of Chan-Vese (white), (c) segmentation via L2S (black) and (d) segmentation via DL2S model (yellow) for the same image. We observe that the segmentation performance of L2S drops significantly for automatic initialization, which is also true for Chan-Vese method. In comparison DL2S has similar segmentation results for both initialization techniques. Quantitative evaluation of performance based on initialization is provided in Table 1.

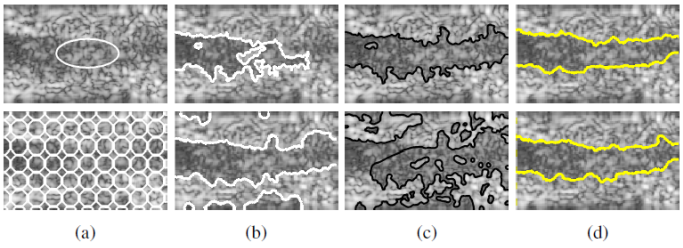


Figure 5: Comparison of segmentation results using manual and automatic initialization methods. (a) initialized contour (b) segmentation results of Chan-Vese (white), (c) segmentation via L2S (black) and (d) segmentation via DL2S model (yellow)

Table 1: Quantitative comparison of the three methods

DL2S [**29**]		Chan-Vese [**9**]		L2S [**30**]	
Manual	Auto	Manual	Auto	Manual	Auto
0.93 ± 0.02	0.92 ± 0.04	0.91 ± 0.07	0.86 ± 0.11	0.89 ± 0.09	0.55 ± 0.17
0.90 ± 0.04	0.90 ± 0.07	0.88 ± 0.12	0.88 ± 0.12	0.90 ± 0.06	0.88 ± 0.12
0.85 ± 0.08	0.86 ± 0.08	0.80 ± 0.08	0.85 ± 0.11	0.85 ± 0.12	0.84 ± 0.09
0.80 ± 0.10	0.83 ± 0.06	0.69 ± 0.21	0.73 ± 0.12	0.70 ± 0.19	0.60 ± 0.14

0.76 ± 0.16	0.76 ± 0.10	0.75 ± 0.14	0.72 ± 0.11	0.72 ± 0.16	0.62 ± 0.13

Quantitative comparison of segmentation: Figure 6 shows the segmentation performance of Chan-Vese (white) [7]), L2S (black) [18] and DL2S (yellow). Fig. 4 shows that DL2S is able to capture the blood vessels more appropriately in presence of severe contrast and intensity inhomogeneity. A quantitative comparison for five datasets is shown in **Error! Reference source not found.**. The Dice index, $\mathcal{D} = \frac{2|s_t \cap s_g|}{\{|s_t|+|s_g|\}}$ is evaluated for the three algorithms. Here s_g denotes the ground truth segmentation and s_t is the segmentation result for DL2S, Chan-Vese or L2S. It is observed that for each dataset, DL2S demonstrates significantly better performance than L2S or CV. Dl2S achieves highest improvement in performance of above 65% in one dataset and 42% in an in-vivo dataset. On average, we observe increase in segmentation accuracy by more than 12% for the all the datasets.

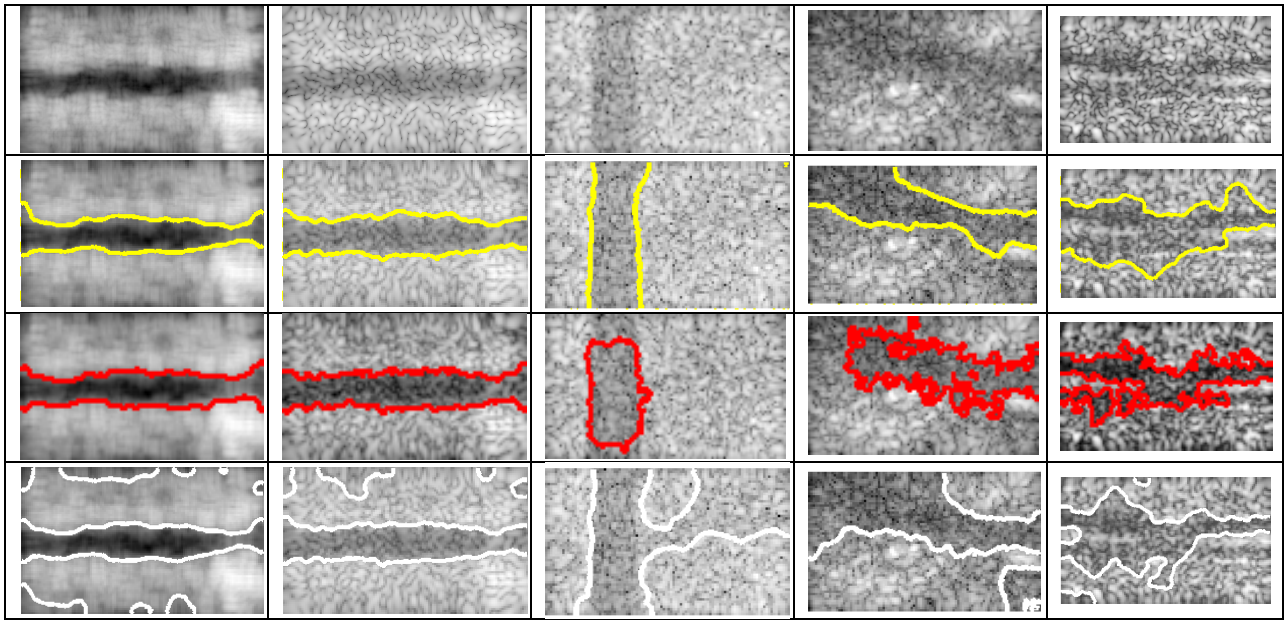


Figure 6: Segmentation comparison of DL2S [**29**] with Chan-Vese [**9**] and L2S [**30**] is shown here. The original C-mode ultrasound images captured with a portable scanner are shown in the first row. Segmentation results of Chan-Vese (white), L2S (black) and DL2S model (yellow) on these images are shown in rows 2, 3 and 4 respectively. The first four images are phantom and the last four are in vivo images of blood vessels

3.2. Image Classification

Here we develop an information theoretic kernel combination method embedded in the dictionary learning framework. One advantage of the kernel space representation, other than a higher-dimensional representation, lies in the fact that different features can be combined in the kernel space using multiple kernel functions [**31**]. The kernel-sparse representation techniques [**15**], [**13**] mainly deal with a single kernel in sparse representation or dictionary learning framework. Our method learns a dictionary in the kernel space for each of the classes in the learning phase. We employ a mutual information based approach to obtain the most desirable weights for kernel combination. In the testing phase, our method exploits the learned dictionaries and the kernel weights to assign a class label to the test. The steps involved in the classification system are shown in the block diagram.

A. Training using multi-kernel dictionary learning

The training phase involves three steps:_The first step involves discriminative feature and respective kernel matrix computation. The next step in training involves mutual information based combination of these features. Finally, the third step is a classifier design using a dictionary learning framework.

a. Discriminative feature and respective kernel matrix computation
b. In image classification, a crucial part involves selecting the most discriminative feature that can differentiate between images belonging to different classes in a dataset. Here we address the feature combination problem by multi-kernel dictionary learning. Different features (as explained in 2.6) are computed for each of the images and the kernel matrix corresponding to that feature are computed. The different kernels used in the experiment are Gaussian ($\mathrm{K}(\mathrm{y_i}, \mathrm{y_j}) = \exp\left(-\gamma \left\|\mathrm{y_i} - \mathrm{y_j}\right\|_2^2\right)$ and polynomial ($\mathrm{K}(\mathrm{y_i}, \mathrm{y_j}) = \mathrm{y_i^T y_j}$), where $\mathrm{y_i}$ and $\mathrm{y_j}$ are the image featuures and γ is a scaling parameter.
c. Mutual information based combination of features
d. This step in the training step involves learning the kernel weights $\mathcal{B}_\mathrm{c} = [\beta_1 \ldots . \beta_\mathcal{S}]$, for each of the classes c = 1….. $\mathcal{C}$. We define the linear combination of the kernel matrix $\mathcal{K} = \sum_{s=1}^{\mathcal{S}} \beta_\mathrm{s} \mathrm{K_s}(.)$, such that $\Psi(.)^\mathrm{T}\Psi(.) = \mathcal{K}$ as explained in 2.3. If f denotes the function transforming the given feature vector to the RKHS, we can re-write the entropy for the transformed feature as $\mathrm{H}(\mathrm{c}|\mathrm{f}(\mathrm{x}))$. The weights for the kernel combination are hence obtained by minimizing the conditional entropy for a class given by the following equation.

$$\begin{aligned} &\underset{\mathcal{B}}{\operatorname{argmin}}\, H(c|\mathcal{K} = f(Y, \mathcal{B})) \\ &\text{s.t } \textstyle\sum_{s=1}^{\mathcal{S}} \beta_\mathrm{s} = 1 \text{ and } \beta_\mathrm{s} \geq 0 \; \forall s \end{aligned} \tag{15}$$

We initialize $\beta_\mathrm{s} = \frac{1}{\mathcal{S}}$ $\forall \mathrm{s} \in [1 \ldots \mathcal{S}]$, such that $\sum_{\mathrm{s}=1}^{\mathcal{S}} \beta_\mathrm{s} = 1$ .To solve for $\mathcal{B}_\mathrm{c}$, we use a random search method [**32**]. We randomly select weight values from a Gaussian distribution such that, $\beta_\mathrm{s}^\mathrm{t} \sim \mathcal{N}(\beta_\mathrm{s}^{\mathrm{t}-1}, 1)$ and normalized by $\sum_{\mathrm{s}=1}^{\mathcal{S}} \beta_\mathrm{s}$. Then select the $\beta_\mathrm{s}$values for which $\mathrm{H}(\mathrm{c}|\mathcal{K} = \mathrm{f}(\mathrm{Y}, \mathcal{B}))$ is minimum. t denotes the iteration step.

e. Multi-kernel dictionary learning: This step involves Updating dictionary $\mathcal{D}$ (as discussed in section 2.2) and sparse codes X for each of the classes separately. With $\mathcal{B}_\mathrm{c}$ is fixed for, $\mathcal{D}_\mathrm{c}$ is initialized by randomly selecting Kcolumns of $\mathrm{Y_c}$. First keeping $\mathcal{D}_\mathrm{c}$ fixed, we update the sparse codes $\mathrm{X_c}$ solving the following. Here $Y = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots\ldots, \boldsymbol{y}_{N_c}]$, $(Y_c \in \mathbb{R}^{n \times N_c})$. The sparse codes for a class can be embedded in the matrix $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots\ldots, \boldsymbol{x}_{N_c}]$, $X_c \in \mathbb{R}^{K \times N_c}$. Let us denote N_c as the number of images in the c^{th} class.

$$\begin{aligned} &\underset{X_c^t}{\operatorname{argmin}} \|\Psi(Y_c) - \boldsymbol{\mathcal{D}}_c^{t-1} X_c^t\|_F^2 \\ &s.t\ \|x_i\|_0 \leq T\ \forall i\ \in 1 \ldots . N_c \end{aligned} \tag{16}$$

We use orthogonal matching pursuit [**33**] to solve (10). Once the sparse codes are obtained the next step is to update the dictionary. Keeping the sparse codes fixed, we solve the following equation, with the constraint that the columns of the dictionary will be orthonormal.

$$\underset{\mathcal{D}_c^t}{\operatorname{argmin}} \|\Psi(Y_c) - \boldsymbol{\mathcal{D}}_c^t X_c^t\|_F^2 \tag{17}$$

The objective function can re-written as follows and the optimized over $\mathbb{D}_c^t$ [**14**].

$$\mathcal{K}(\boldsymbol{y}_i, \boldsymbol{y}_i) - 2\boldsymbol{x}_i^T \boldsymbol{\mathcal{D}}^T \mathcal{K}(\boldsymbol{Y}, y_i) + \boldsymbol{x}_i^T \boldsymbol{\mathcal{D}}^T \mathcal{K}(Y, Y) \boldsymbol{\mathcal{D}} \boldsymbol{x}_i \tag{18}$$

We use the K-SVD algorithms [**10**] for the dictionary update.

B. Testing phase

The testing part involves extracting similar types of features as used for training part and using the learned dictionary to identify the class for the test image

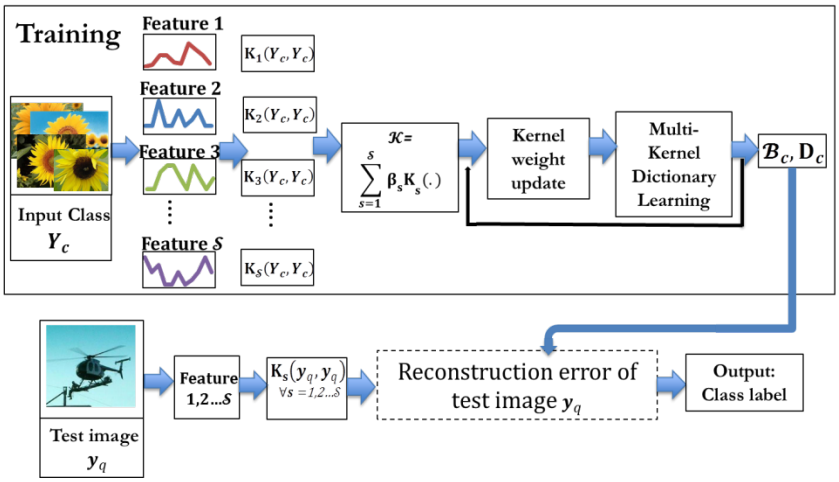


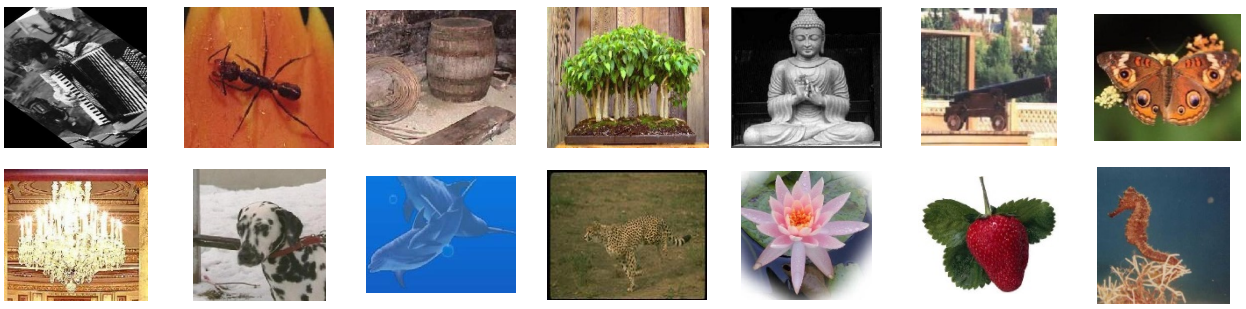
Figure 7: Block diagram of the classification method showing the stages of training and testing phase

The classification of the test image is performed based on the minimum reconstruction error with respect to the class dictionaries. Once the feature vectors for the query image, $\boldsymbol{y}_q$ is available, $\forall c = 1 \ldots . . \mathcal{C}$, the kernel combination for the test image is obtained as $\mathcal{K}_c = f(\boldsymbol{y}_q, \mathcal{B}_C)$, such that $\mathcal{K}_c = \Psi_c^T \Psi_c$. The respective sparse codes $\boldsymbol{x}_q^c$ corresponding to the class dictionary is obtained by solving the following

$$\underset{x_q}{\operatorname{argmin}} \left\|\Psi_c(\boldsymbol{y}_q) - \mathcal{D}_c \boldsymbol{x}_q^c\right\|_2^2 \; s.t. \left\|\boldsymbol{x}_q^c\right\|_0 \leq T \tag{19}$$

The test image is identified to belong to the particular class for which the reconstruction error is minimum given by the following, $(\ell(\boldsymbol{y}_q) = c\ ) = \min_c \left\|\Psi_c(\boldsymbol{y}_q) - \mathcal{D}_c \boldsymbol{x}_q^c\right\|_2^2$

The details of the algorithm are provided in Table 2.



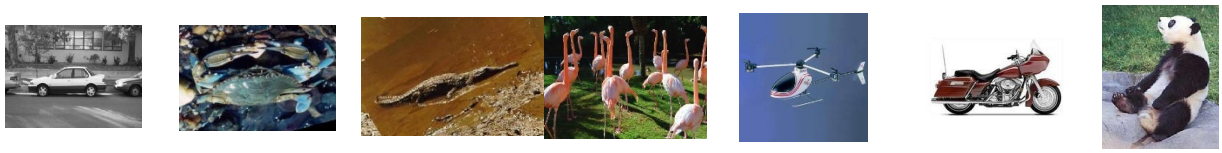


Figure 8: Sample images from Caltech 101 dataset

Experimental Results

We performed experiments on Caltech 101 dataset [**34**]. The dataset has 101 categories with about 9000 images. About 3000 images were used for training. 30 images were chosen at random per class to perform the training. The rest, about 6000 images were used for testing. Sample images from the dataset are shown in Figure 8. We performed the experiment with spatial pyramid [**35**] representation of scale invariant feature transform (SIFT) descriptors [**36**], with Gaussian (and polynomial and histogram of oriented gradients (HOG) [**22**] descriptors with polynomial kernel. In Figure 10, we plot the classification accuracy $\left(\frac{\#no.of\ images\ classified\ correctly}{\#no.of\ test\ images}\right)$ of 70 classes in ascending order of their classification accuracy. Figure 9 shows comparison of classification accuracy of MI-MKDL with meta-algorithm for feature nomination [**3**] of 24 sample classes. MI-MKDL shows on average an increase of 10% in classification accuracy with respect to the meta-algorithm

Table 2: Mutual Information based multi-kernel dictionary learning algorithm

Training Step:
Input: Training data, $Y = [Y_1, Y_2, \ldots\ldots, Y_C]$,
$\mathcal{B} = [\beta_1 \ldots . \beta_S]$ Initialize $\beta_s = \frac{1}{S}\ \forall s \in [1 \ldots S]$.
Output: For each class c, $\mathcal{B}_c, X_c, \mathcal{D}_c$
Method: For class c = 1….. $\mathcal{C}$,
Set $t = 1$
(i) Compute kernel matrix $\mathcal{K} = \sum_{s=1}^{S} \beta_s^t K_s(Y_c, Y_c)$
(ii) Solve for $\mathcal{B}_c^t$, that minimizes conditional entropy.

$$\underset{\beta}{\operatorname{argmin}}\, H(c|f(Y, \mathcal{B}^t))$$

$$\text{s.t. } \sum_{s=1}^{S} \beta_s = 1 \text{ and } \beta_s \geq 0\ \forall s$$

(iii) Multi-kernel dictionary learning
a. Sparse coding stage, given $\mathcal{D}_c^{t-1}$,

$$\underset{X_c^t}{\operatorname{argmin}} \|\Psi(Y_c) - \mathcal{D}_c^{t-1} X_c^t\|_F^2$$

$$s.t\ \|x_i\|_0 \leq T\ \forall i\ \in 1 \ldots . N_c$$

b. Dictionary update step, given X_c^t,

$$\underset{\mathcal{D}_c^t}{\operatorname{argmin}} \|\Psi(Y_c) - \mathcal{D}_c^t X_c^t\|_F^2$$

(iv) Set $t = t + 1$ until convergence is reached
Testing Step:
Input: Test data $\boldsymbol{y}_q$, $\mathcal{B}_c, \mathcal{D}_c\ \forall c = 1 \ldots C$
Output: Class label $\ell(\boldsymbol{y}_q)$
Method:
(i) Solve for sparse codes for the test image:

$$\underset{x_q}{\operatorname{argmin}} \left\|\Psi_c(\boldsymbol{y}_q) - \mathcal{D}_c \boldsymbol{x}_q^c\right\|_2^2\ s.t. \left\|\boldsymbol{x}_q^c\right\|_0 \leq T$$

(ii) The class label is assigned as

$$(\ell(\boldsymbol{y}_q) = c\ ) = \min_c\ \left\|\Psi_c(\boldsymbol{y}_q) - \mathcal{D}_c \boldsymbol{x}_q^c\right\|_2^2$$

4. Conclusion and Future direction

Here we have proposed two segmentation methods, a learning based threshold selection method and a learning based intensity modeling method. Moreover we have also proposed mutual information based adaptive feature combination method for kernel dictionary learning.

In the learning-based thresholding method for grayscale image segmentation, the image histogram is first smoothed and normalized by a Gaussian kernel and a best fit by a Gaussian mixture to the smoothed histogram is then sought through a learning process. In particular, the parameters (mean, standard deviation, and weight) associated with each component in the Gaussian mixture are estimated with a set of learning automata. Thresholds are computed based on these parameter estimates.

For future work, we propose to use a multivariate Gaussian mixture as a model of the high-dimensional color feature distribution (possibly combined with texture features and spatial

information) for image retrieval. Unlike existing image retrieval schemes using modified EM algorithms, we will extend the proposed learning automaton approach to cope with multivariate Gaussian parameter estimation for color image segmentation. Detected image components can then be indexed and matched to a query image for retrieval purposes.

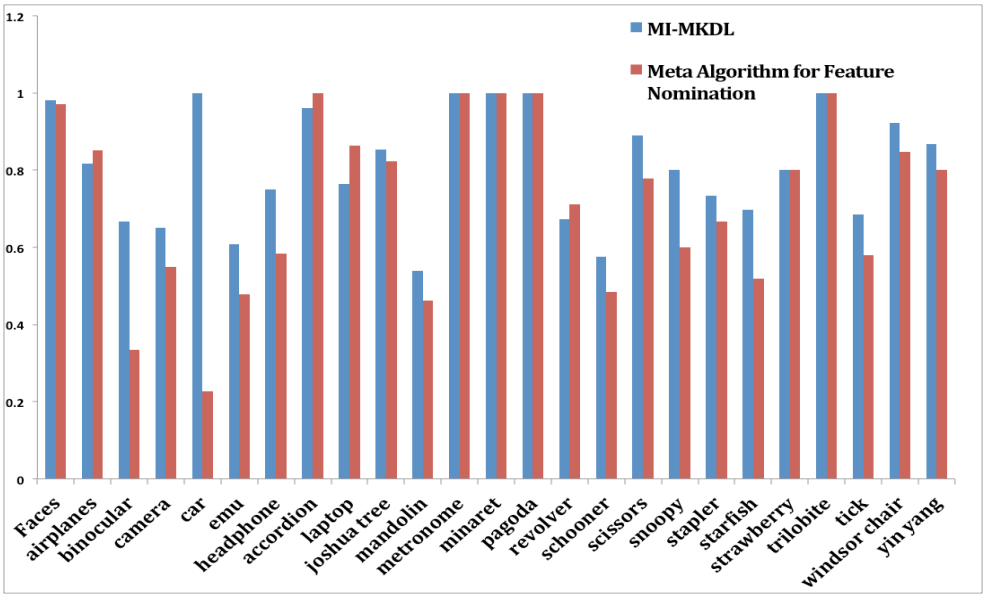


Figure 9: Classification accuracy for 70 classes from Caltech 101 dataset

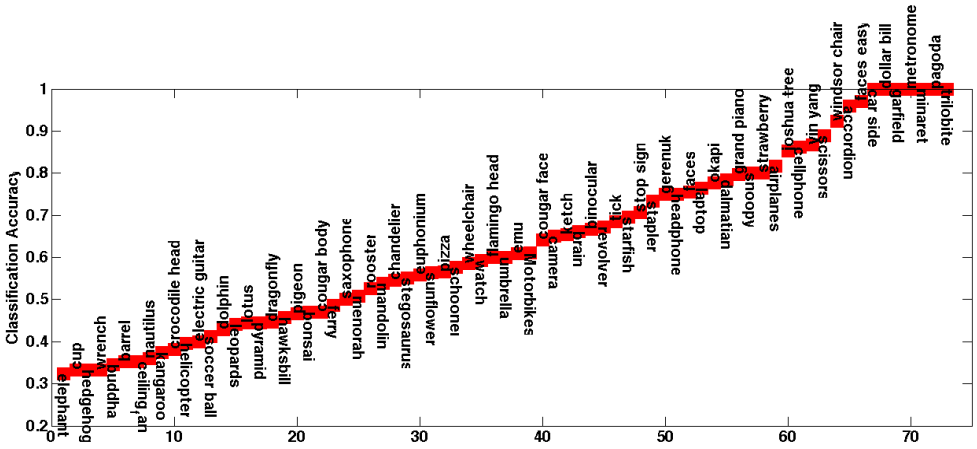


Figure 10: Classification accuracy comparison of MI-MKDL and meta-algorithm for feature nomination

In the second approach a novel segmentation method is proposed which combines the idea of dictionary learning and region based segmentation algorithm in presence of significant clutter and heterogeneous intensity. Furthermore, it has been shown using quantitative and qualitative results that the proposed method outperforms the state of the art in terms of contour initialization and demonstrates accurate segmentation in cluttered images without the use of explicit shape priors. The proposed method requires a set of images with similar objects for the dictionary learning, which may not always be available.

For future work we propose to learn a dictionary from image patches and employ them to model the different regions of the image for segmentation method.

In our third approach, we present a method of feature selection and combination for robust image classification. We first introduce a sparse representation based dictionary learning algorithm using kernel space feature representation and then extend this representation by way of multi-kernel learning. The multi-kernel learning allows feature combination and the feature selection is optimized using mutual information, yielding weights for kernel combination. In these preliminary experiments, our method shows an average increase of 10% in classification accuracy over that achieved by the meta-algorithm for feature nomination.

References

[1] M. N. Howell and T. J. Gordon, "Continuous action reinforcement learning and their application to adaptive digital filter design," , Engineering Applocations of Artificial Intelligence.

[2] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE transactions on Image processing*, vol. 10, no. 2, pp. 266-277, 2001.

[3] R. Sarkar, K. Skadron, and S. T. Acton, "A Meta-Algorithm For Classificatiob by Feature Nomination," in *IEEE International Conference on Image Processing*, 2014.

[4] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust Face Recognition via Sparse Representation," *IEEE Transactions on PAMI*, vol. 31, no. 2, pp. 210-227, 2009.

[5] Z. Jiang, Z. Lin, and L. S. Davis, "Label Consistent K-SVD: Learning a Discriminative Dictionary for Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651 - 2664, 2013.

[6] Q. Zhang and B. Li, "Discriminative k-svd for dictionary learning in face recognition," , 2010, IEEE Conference on Computer Vision and Pattern Recognition.

[7] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *ICCV*, 2011.

[8] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communicationstions on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577-685, 1989.

[9] T.F. Chan and L.A. Vese, "Active contour without edges," *IEEE Transaction on Image Processing*, vol. 10, no. 2, pp. 266-277, 2001.

[10] M. Elad and M Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *Image Processing IEEE Transactions on*, vol. 15, no. 12, pp. 3736-3745, 2006.

[11] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," *Proceedings of the 26th Annual International Conference on Machine Learning,ACM*, 2009.

[12] J. Yang, A. F. Frangi, J. Y. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 230-244, 2005.

[13] S. Gao, I. W. H. Tsang, and L. T. Chia, "Kernel sparse representation for image classification and face recognition," in *ECCV* , 2010.

[14] H. V. Nguyen and et al., "Kernel dictionary learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012.

[15] L. Zhang et al., "Kernel sparse representation-based classifier," *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1684-1695, 2012.

[16] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on, PAMI*, vol. 27, no. 8, pp. 1226-1238., 2005.

[17] M., Vasconcelos and N. Vasconcelos, "Natural image statistics and low-complexity feature selection," *Pattern Analysis and Machine Intelligence*, vol. 31.2, pp. 228-244, 2009.

[18] François Fleuret, "Fast binary feature selection with conditional mutual information," *The Journal of Machine Learning Research* , vol. 5, pp. 1531-1555., 2004.

[19] Z. Wang, Q. Zhao, D. Chu, F. Zhao, and L. J. Guibas, "Select informative features for recognition," in *ICIP*, 2011.

[20] Y. Grandvalet and Y Bengio, "Semi-supervised Learning by Entropy Minimization," in *Neural Information Processing Systems*, 2005.

[21] M. Elad and M Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *Image Processing, IEEE Transactions on*, vol. 15(12), pp. 3736-3745., 2006.

[22] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[23] K. S. Narendra and M. A. L. Thatcher, *Learning Automata: An Introduction*. NJ, USA: Prentice Hall, Inc., 1989.

[24] Q. Sang, Z. Lin, and S. T. Acton, "Learning Automata for Image Segmentation," *Pattern Recognition Letters (submitted)*, 2015.

[25] M.J. Carlotto, "Histogram analysis using a scale-space approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence* , vol. 9, pp. 121-129, 1987.

[26] H. Beigy and M. R. Meybodi, "A new continuous action-set learning automaton for function optimization," *Journal of Franklin institute, (*, vol. 343, pp. 27-47, 2006.

[27] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, 2nd ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co, 1992.

[28] M. Sonka, V, Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision*.: Thomson-Engineering, 2007.

[29] R. Sarkar, S. Mukherejee, and S. T. Acton, "Dictionary Learning Level Set," *Signal Processing Letters (Submitted)*, 2015.

[30] S. Mukherjee and S. T. Acton, "Region Based Segmentation in Presence of Intensity Inhomogeneity Using Legendre Polynomials," *Signal Processing letters*, vol. 22, no. 3, pp. 298 - 302, 2015.

[31] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *IEEE 12th International Conference on Computer Vision*, 2009.

[32] H. Hino, N. Reyhani, and N. Murata, "Multiple kernel learning by conditional entropy minimization," in *IEEE Ninth International Conference on Machine Learning and Applications*, 2010.

[33] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53.12, pp. 4655-4666, 2007.

[34] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few trainig samples an incremental Bayesian approach tested on 101 object categories," in *CVPR, Workshop on Generative-Model based vision*, 2004.

[35] S Lazebnik, C Schmid, and J Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.

[36] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91-110, 2004.