Feature Selection for Appearance-based Vehicle Tracking in Geospatial Video

Mahdieh Poostchi[a] Filiz Bunyak[a] Kannappan Palaniappan[a] Guna Seetharaman[b]

[a]Department of Computer Science, University of Missouri-Columbia
[b]Air Force Research Laboratory, Rome, NY, USA

ABSTRACT

Current video tracking systems often employ a rich set of intensity, edge, texture, shape and object level features combined with descriptors for appearance modeling. This approach increases tracker robustness but is computationally expensive for realtime applications and localization accuracy can be adversely affected by including distracting features in the feature fusion or object classification processes. This paper explores offline feature subset selection using a filter-based evaluation approach for video tracking to reduce the dimensionality of the feature space and to discover relevant representative lower dimensional subspaces for online tracking. We compare the performance of the exhaustive FOCUS algorithm to the sequential heuristic SFFS, SFS and RELIEF feature selection methods. Experiments show that using offline feature selection reduces computational complexity, improves feature fusion and is expected to translate into better online tracking performance. Overall SFFS and SFS perform very well, close to the optimum determined by FOCUS, but RELIEF does not work as well for feature selection in the context of appearance-based object tracking.

Keywords: Feature Selection, Object Tracking, SFFS, SFS, FOCUS, RELIEF, Geospatial Video

1. INTRODUCTION

Object tracking in video requires robustness to imaging conditions, environmental characteristics, sensor response and appearance variability. Current video tracking systems often employ a rich set of intensity, edge, texture, shape and object level features combined with descriptors for appearance modeling.[1–11] These descriptors are used in conjunction with other cues such as motion, object class, and background clutter to detect and track targets over time. This approach increases tracker robustness but is computationally expensive for realtime applications and localization accuracy can be affected by including lower quality features in the feature fusion or object classification processes. This paper explores offline feature subset selection for video tracking to reduce the dimensionality of the feature space and to discover a representative lower dimensional (non-projection) subspace for online tracking. Optimal feature subset selection is combinatorially intractable when the feature space is large. We compare the complete FOCUS[12] to the dynamic sequential floating forward search (SFFS),[13] the greedy sequential forward selection (SFS)[14] and the instance learning based RELIEF[15] models - four standard approaches used in machine learning. Given an application specific evaluation function, the FOCUS algorithm exhaustively explores the best feature combination, while the SFFS algorithm sequentially finds the local best feature subset by considering the conditional inclusion and exclusion of the features. The SFS is a greedy algorithm and may fail to select the optimal solution. The RELIEF algorithm doesn't directly select the best features but rather gives each feature a weight indicating its level of relevance to the class label.

Our feature subset evaluation system is independent of the full tracking environment and uses just the ground truth target locations. We developed a separate test-bed for filtering-based feature selection in order to decouple feature performance from the rest of the tracking system where the final outcome depends not only on the features used but also on the other parameters like the predictor performance and target kinematics. Likelihood maps for each feature are constructed using sliding window comparison between the target and a region of interest (ROI). Local maxima in each feature likelihood map are sorted based on their peak strengths (match likelihoods). The peak rank/order of the local maxima corresponding to the target location is used to quantify the performance of the feature producing the likelihood map. Several likelihood fusion methods can be used to combine multiple feature likelihood maps into a single joint likelihood map at each iteration. Each of the feature sets is evaluated over all the targets and frames to obtain an aggregate score. All four feature selection algorithms used in this

Geospatial InfoFusion III, edited by Matthew F. Pellechia, Richard J. Sorensen,
Kannappan Palaniappan, Proc. of SPIE Vol. 8747, 87470G · © 2013 SPIE
CCC code: 0277-786X/13/$18 · doi: 10.1117/12.2015672

paper are evaluated under the same conditions. FOCUS which is not practical for larger feature sets because of its exhaustive search, is feasible in this study and produces the best results. The SFFS-based selection has performance similar to greedy SFS selection and both outperform the RELIEF method for the vehicle tracking application. The linear RELIEF unlike the other feature selection methods which produce the same results each time, uses random class sampling so the resulting weights change from run to run.

Experiments show that using offline feature selection reduces computational complexity, improves feature fusion that can in turn lead to better online tracking performance. Overall SFFS and SFS perform very well, close to the optimum determined by FOCUS, but RELIEF does not work as well for feature selection in the context of appearance-based object tracking. Section 2 presents a short review of our interactive low frame rate Likelihood of Features Tracking (LoFT) system. Section 3 describes the four well-known feature selection algorithms used in this study. Section 4 describes the evaluation testbed followed by experimental results and conclusions.

2. FEATURE SET FOR APPEARANCE-BASED VEHICLE TRACKING IN WAMI

Wide area motion imagery (WAMI) systems enable persistent surveillance of wide regions of interest using orbiting airborne platforms for geospatial applications.[2,16] But the large volume of visual imagery from a continuously varying collection of viewing angles obtained by these systems poses unique challenges including accurate registration,[10,17] severe appearance changes due to changing camera viewing angle and target pose, large object displacements caused by low frame rate sampling, low resolution targets and static or dynamic occlusions.[1–4] Recently we have developed an appearance-based Likelihood of Features Tracking (LoFT) system, specialized for low frame rates and large object displacements, that uses a rich set of features and feature fusion to address the specific challenges of WAMI data.[1,3,18,19] Large data handling , motion imagery visualization and track visualization are supported.[20–25] The search window for the target is set as a small subwindow or region of interest centered at the predicted target position obtained using the Kalman or particle filters. Matching likelihood maps for individual features are computed using normalized cross correlation or sliding window histogram similarity operators (Figure 1). The integral histogram method[26–28] is used to accelerate computation of the sliding window histograms for *a posteriori* likelihood estimation.

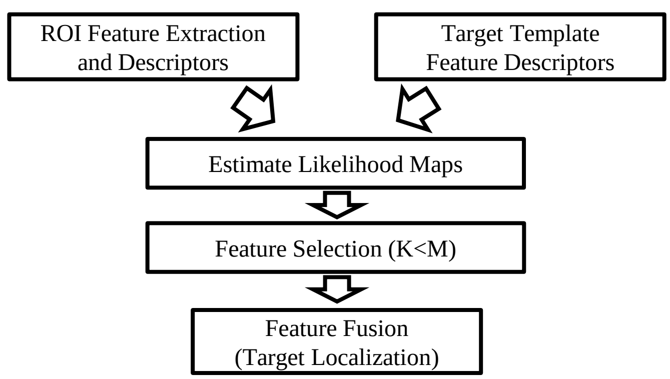


Figure 1. Sliding window matching likelihood computation between template and local ROI.

LoFT includes a large set of low level image-based feature descriptors that account for color, texture and contour/edge properties of target objects; in the current work we used eleven features as shown in Figure 2. Joint likelihood maps are obtained by fusing individual feature likelihood maps, assuming independence among features. The feature likelihood maps are computed using the sliding window comparison methods between the target and ROI feature histograms. Figure 1 shows how the feature likelihood maps are constructed (by target to search window similarity matching), selected and fused. Intensity, gradient magnitude and shape index normalized cross-correlations, INCC, GMNCC, and SINCC respectively, are the region-based similarity measures that incorporate spatial information. These three likelihood maps are computed using pixel-based cross correlation using integral image methods, and provide good spatial localization and discriminative power but are sensitive to changes in pose, viewing angle and scale. Histogram-based descriptors on the other hand provide global information about an object that can be robust to changes due to motion, pose, or viewing angle using appropriate normalization and alignment operations. Gradient histograms further increase this robustness with decreased sensitivity to illumination change. LoFT uses six histogram based features including three gradient-based (first derivative) operators (HMG, HOG, HARST) and three Hessian matrix-based (second derivative) operators (HSI, HNCI, HEO) as shown in Figure 2. The adaptive robust structure tensor (ARST) operator provides more accurate local orientation estimation in the presence of noise.[29] The shape index and the normalized curvature index are derived from the eigenvalues of the Hessian matrix. The magnitude weighted histogram of the Hessian eigenvector orientations are used to compute local shape-based features. In addition to these feature descriptors two additional descriptors are used – intensity histogram (HI) to represent intensity/color information, and the local binary pattern (HLBP) histogram to capture statistical and structural properties of object texture.

Each of the feature maps is binned into ten categories except for HLBP. LoFT uses the uniform rotation-invariant LBP consisting of 18 pattern classes.[1,3,30] The dimensionality of the feature space for feature selection can be interpreted in several different ways – if the output of each feature operator is treated as a feature vector and we concatenate pixel level information within the car template model window then the dimensionality of the feature space would be $13,475$ for a template model window size of 35×35 pixels. On the other hand if the pixel level information is aggregated into region statistics and we concatenate the histogram bins together then the dimensionality of the feature space will be only 118. If we use the feature likelihood maps after histogram-based matching then the dimensionality will be just 11. In this paper we use the last case in order to be able to compare results with the exhaustive enumeration approach (*i.e.* FOCUS) which finds the optimal feature subset. Note the combinatorial enumeration using FOCUS for feature subset selection is not possible for the other two cases since the feature space dimensionality is too large.

Candidate Feature Set	Feature Label	Feature Number	H/C
Intensity histogram	HI	1	H
Intensity normalized cross correlation	INCC	8	C
Gradient Magnitude	HMG	2	H
Gradient Orientation	HOG	6	H
ARST Orientation	HARST	10	H
Gradient Magnitude normalized cross correlation	GMNCC	9	C
Shape Index	HSI	3	H
Normalized Curvature Index	HNCI	4	H
Hessian eigenvector orientations	HEO	5	H
Shape Index correlation	SINCC	11	C
LBP	HLBP	7	H

Figure 2. Visual appearance-based feature descriptors used in LoFT for object tracking in WAMI.[1,3]

3. BRIEF OVERVIEW OF SELECTED FEATURE SELECTION METHODS

Non-discriminative features not only degrade system performance in terms of consuming additional computational resources, but also decrease target localization performance. Given a set of candidate features, the main goal of feature selection is to find the minimal number of features that achieves the best performance in terms of a filtering evaluation criteria. Feature selection also has the additional benefit of speeding up performance since fewer feature computations need to be performed.[15,31,32] Feature selection algorithms facilitate data visualization, improve computational performance, and increase adaptability and flexibility.[33] Feature selection models with different evaluation measures are generally categorized into filter models such as SFS,[14] SFFS,[13] FOCUS,[12] RELIEF,[15] wrapper models like SBS-SLASH,[34] W-SBG/W-SFG,[35] ELSA[36] and hybrid models like BBHFS,[37] Xings's.[38] In the filter models, the feature selection is a preprocessing step where feature subsets are ranked and selected based on the general characteristics of the data, whereas the wrapper methods evaluate the feature subsets according to a given predictor[31,33,39] and the hybrid models take advantages of the two models. The goal of this paper is to choose a small subset of features that is necessary and sufficient enough to represent the target concept. In this paper the performance of the three well known sequential search strategies SFS, SFFS and RELIEF are compared to the exhaustive FOCUS method.

3.1 Sequential Forward/Backward Selection

The sequential forward selection (SFS) and the sequential backward selection/elimination (SBS) are variations of greedy hill-climbing approaches.[39] SFS starts from the empty set and sequentially adds feature x^+ that improves the objective function the most when combined with the current selected feature set. The backward counterpart starts with the full feature set and at each step sequentially discards the worst performing feature. These greedy methods do not guarantee the optimal solution since there is no chance to change the nested feature subsets in later steps and the global optimal subset may reside in a region far away from the visited search space.[13,14,31]

Algorithm 1 Sequential Forward Selection(SFS)[13,14,31]

Input : $J(.)$- Evaluation measure (accuracy maximizing)
$S(X)$- Candidate feature set
Output : X^*- Selected feature subset
$X_0^* = \{\}, k = 0$
repeat
$x^+ = \underset{x \notin X_k}{argmax} J(X_k^* + x)$
$X_{k+1}^* = X_k^* + x^+; k = k + 1;$
until no improvement in J in the last j steps
or $X^* = S(X)$

Algorithm 2 Sequential Backward Selection(SBS)[13,14,31]

Input : $J(.)$- Evaluation measure (accuracy maximizing)
$S(X)$- Candidate feature set
Output : X^*- Selected feature subset
$X_0^* = S(X), k = 0$
repeat
$x^- = \underset{x \in X_k}{argmax} J(X_k^* - x)$
$X_{k+1}^* = X_k^* - x^-; k = k + 1;$
until no improvement in J in the last j steps
or $X^* = \{\}$

3.2 Sequential Floating Forward/Backward Selection

SFFS/SFBS is a sequential feature selection procedure with a dynamically adaptive number of forward/backward steps. SFFS starts by selecting the best individual feature candidate. Then each feature inclusion (+1 forward) step is followed by a variable number, possibly null, of feature exclusion ($-r$ backward) steps. The exclusion steps continue as long as the generated subsets result in better performance than the previous best subset obtained so far. The SFFS gives up completeness and converges to the closest local optimum, by floating around a potentially good solution search space region. The computational cost may grow exponentially due to the maximum level of backtracking.[13,31] The SFFS doesn't require a monotonic evaluation criterion (this means that the value of the evaluation criterion does not decrease by adding a new feature to the current set). The maximum level of backtracking can be controlled to avoid excessive computations.[13]

3.3 RELIEF

RELIEF is a feature selection algorithm inspired by instance-based learning.[15] It uses a statistical method rather than heuristic search used in the previous methods. RELIEF does not directly generate a subset of candidate features, but rather it gives each feature a weight that indicates its level of relevance to the class label. At each iteration a random instance s is selected from the set of all samples S and updates a feature weight vector W by calculating the distances from s to the nearest hit and the nearest miss. The nearest hit is the closest instance s among all the instances in the foreground class of S and the nearest miss is the closest instance to the nearest background class among all instances of S. Finally, those features whose relevance level is above the desired threshold λ[15,31,40] can be selected. From the theoretical perspective, the relevance level is positive when the feature is relevant and close to zero or negative when it is less relevant. The RELIEF approach is noise tolerant and relatively fast.[15] It requires linear time in the number of given features and the number of training instances, regardless of the target concept to be learned.

Algorithm 3 Sequential Floating Forward Selection(SFFS)[13,31]

Input : $J(.)$- Evaluation measure (accuracy maximizing)
$S(X)$- Candidate feature set
Output : X^*- Selected feature subset
Initialization:
$X_0^* = \{\}$, $k = 0$
(in practice one can begin with k=2 by applying SFS twice)
Termination:
Stop when k equals the number of the features required
Step 1: Inclusion (select the best feature)
$x^+ = \underset{x \notin X_k^*}{argmax} J(X_k^* + x)$
$X_{k+1}^* = X_k^* + x^+;\ k = k + 1;$
Step 2: Exclusion (select the worst feature)
$x^- = \underset{x \in X_k^*}{argmin} J(X_k^* - x)$
if $J(X_k^* - x^-) > J(X_k^*)$ **then**
$X_{k-1}^* = X_k^* - x^-;\ k = k - 1;$
Go to step 2
else
Go to step 1
end if

3.4 FOCUS

The FOCUS algorithm is an exhaustive combinatorial feature selection approach that guarantees finding the optimal subset of features among all 2^n subsets of the n dimensional feature space. This combinatorial feature selection method searches all feature subsets to determine the smallest subset of candidate features that provides the best performance of the filtering criterion. It starts from selecting the best singleton feature set. Then from the subset of size two and so forth until it hits the threshhold.[12,31,39,41] Algorithm 5 describes the steps in the FOCUS feature selection method.

Algorithm 4 RELIEF Algorithm[15,31]

Input : d - distance measure
p - sampling percentage
$S(X)$ - a sample S described by $X, |X| = n$
Output : W- array of feature weights
Set $W[] = 0$;
for $i = 1$ to $p|S|$ **do**
I= randomly select an instance (S);
find nearest hit I_{nH} and nearest miss I_{nM} to I;
for $j = 1$ to n **do**
$W[j] = W[j] + d_j(I, I_{nM}) - d_j(I, I_{nH})$
end for
end for

4. FEATURE SELECTION TEST-BED USING TRACKING CONTEXT

In order to evaluate the feature selection module independently from the rest of the tracking system (i.e. prediction and update modules) and from the target kinematics, we have developed a feature subset evaluation testbed. The proposed test-bed performs three tasks: (1) computes individual likelihood maps for each feature;

Algorithm 5 FOCUS Algorithm[12,31]

Input : $J(.)$- Evaluation measure (accuracy maximizing)
J_0- Minimum allowed value of J
$S(X)$- Candidate feature set
Output : X^*- Selected feature subset

```
for i ∈ [1..n] do
  for each X* ⊂ X, with |X*| = i do
    if J(S(X*) > J_0 then
      STOP
    end if
  end for
end for
```

(2) construct fused likelihood maps for the selected feature subsets; and (3) evaluate feature subsets using the filtering method based on the fused likelihood map.

In order to decouple feature evaluation from the rest of the tracking system, at each frame t, the search window for the target is set to an $m \times m$ region around the true target ground truth position for that frame (instead of the predicted target position in the LoFT tracking system). The performance of the feature subsets can then be readily evaluated since the true target location is known and the search region is the same for all feature subsets. Likelihood maps for individual features are computed using normalized cross correlation or sliding window histogram distances. Joint likelihood maps are constructed by fusing individual likelihood maps using weighted sums. Equal weight fusion is used in this study in order to minimize the influence of likelihood fusion approach on feature selection performance which is described elsewhere.[3,18,19] The joint likelihood map for feature subset X^* for frame t is estimated as:

$$L_{X^*}(t) = \sum_{i=1}^{\mathrm{card}(X^*)} w_i \; L_{X_i^*}(t), \quad w_i = \frac{1}{\mathrm{card}(X^*)} \tag{1}$$

where $L_{X^*}(t)$ is the fused likelihood map for feature subset X^* at time t, $L_{X_i^*}(t)$ is likelihood map for feature i and $card(X^*)$ is the number of features in the subset. Other feature fusion methods such as variational-ratio, distracter index, feature prominence[18] and Chernoff Information[42] can be used as part of the feature selection test bed or in the actual LoFT tracking system.

The filtering score for a feature subset is determined by the target localization of its corresponding likelihood map. The likelihood map scoring is done as follows. Likelihood maps typically contain a number of peaks/local maxima. The height of a peak $L(p,t)$ is the likelihood that the target is located at peak position p. In the ideal case the highest score for a feature set is when the most likely (highest) peak in the corresponding likelihood map is located on the target (*i*.e. zero distance to target). The scoring process ranks peaks in the fused likelihood map in decreasing order of their heights. The highest peak is labeled as rank 1 and higher ranks are assigned to the other lower confidence peaks (Figure 3). Once the peaks are ranked, the score of a likelihood map $L(t)$ is determined by the rank of the highest peak inside the target ground truth region, or is penalized as a miss if there is no local peak present in the target region,

$$score(L(t)) = \begin{cases} rank\left(\arg\max_{p \in R_{GT}}(L(p,t))\right) & \text{if} \quad \exists p \in R_{GT} \\ k+1 & \text{otherwise} \end{cases} \tag{2}$$

The likelihood map scoring process is illustrated in Figure 4. The score of a feature subset X^* is computed as the average likelihood score over the total number of processed frames where occluded frames are ignored.

$$score(X^*) = \frac{\sum_{t=1}^{\mathrm{card}(frames)} score(L_{X^*}(t))}{\mathrm{card}(frames)} \tag{3}$$

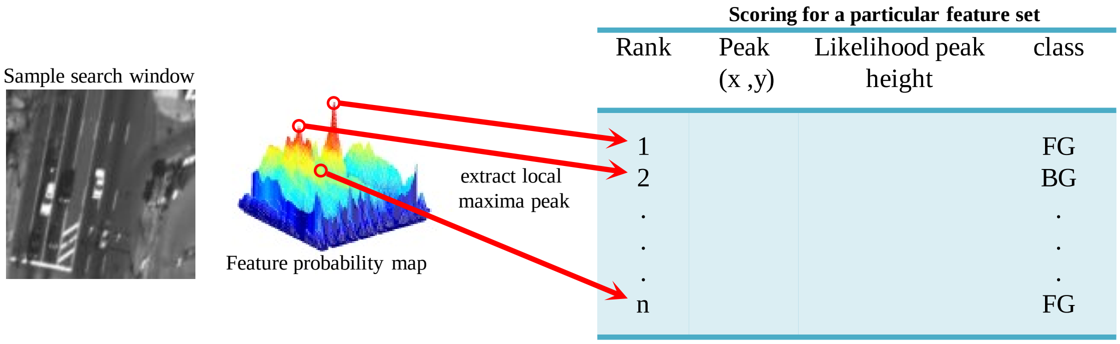
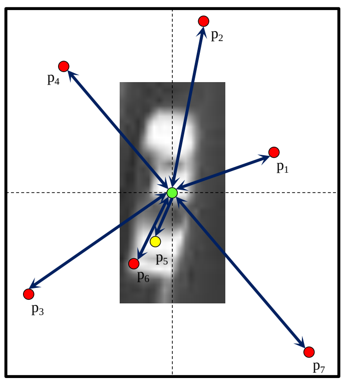


Figure 3. Evaluation of a fused match likelihood map produced by a feature set. From Left to right: search window, target to sliding window match likelihood map, local maxima (peaks) in the likelihood map and corresponding rank, position and class information.



Rank	Peak (x ,y)	Likelihood peak height	class
1	p_1	$L(x_1, y_1)$	BG
2	p_2	$L(x_2, y_2)$	BG
3	p_3	$L(x_3, y_3)$	BG
4	p_4	$L(x_4, y_4)$	BG
5	p_5	$L(x_5, y_5)$	FG
6	p_6	$L(x_6, y_6)$	FG
7	p_7	$L(x_7, y_7)$	BG

Figure 4. An example for scoring process. Among the observed local maxima in the match likelihood map, two of the local maxima fall within the target region, p_5 and p_6. The index of the lowest ranked peak in the target region, 5 in this case, is used as the score of the feature subset. Lower scores indicate better feature sets.

5. EXPERIMENTS

5.1 The PSS Imaging Array Characteristics

In this paper we used WAMI Persistent Surveillance Systems (PSS) imagery acquired from an eight camera array for Philadelphia. Each camera in the array produces an 11 megapixel 8-bit gray scale image typically 4096×2672 at one to four frames per second.[1] These raw images are georegistered to a 16384×16384 image mosaic with a ground sampling distance of about $25cm$ for the imagery used in this paper (for more details of the optical characteristics of the camera array imaging system and processing challenges refer to[2]). Experiments and the feature performance evaluation are performed on fifteen selected PSS cars corresponding to different appearance and environment complexities.

5.2 Implementation of Feature Selection Methods

We implemented FOCUS, SFFS and the SFS feature selection methods using matlab environment and integrated them to the LoFT feature selection module. These three methods directly use the scoring scheme discussed in Section 4. For RELIEF a feature selection matlab toolbox has been used.[43] We couldn't use the ranked features directly for RELIEF due to its feature weight updating procedure. Therefore, we constructed a feature value data set for each of the fifteen PSS cars. Each row of the RELIEF PSS car training data set has 12 columns. The first 11 columns correspond to the 11 feature likelihood values and the last column shows the target class label(FG=+1, BG=-1). In many two class learning algorithms we often test with classifiers that support a -1 and +1 class labeling such as Support Vector Machines. So it is not unusual to use the class label -1 for background and +1 for foreground (as we did for the other feature selection methods(SFFS, SFS, FOCUS)). However, it is important to note that the implementation of RELIEF[43] requires positive class labels (ie 1 for background, 2 for foreground), otherwise the results will be incorrect. Also, unlike the other feature selection methods which are deterministic and produce the same results at each run, RELIEF uses random class sampling so the weights change from run to run.

5.3 Experimental Results

We evaluated LoFT feature set performance on the fifteen PSS cars over a total of 2290 frames. Figure 5 shows the results of the selected five vehicles with different characteristics. The observed results illustrate that there are many factors that affect the performance of a particular feature. Image resolution, target size, target color (i.e. light or dark car) and background complexities change from one car to the other car. The obtained results validate that the intensity and the gradient magnitude (both histogram and correlations) are the most discriminative features for the light cars. While in the case of dark cars, additional feature descriptors like HOG, HOE and LBP are required. Therefore, different number and type of feature descriptors are required to accommodate the vehicle appearance changes. In order to obtain general results for all cars with different characteristics, we construct a pool of all PSS cars. Figures 6 and 7 compare the performance of four different feature selection methods and give us a general perspective on the quality of the various LoFT features. FOCUS does complete search and therefore gives the optimal subset. SFFS and SFS reach close to the local optimal subset, but don't necessarily find the best solution. RELIEF does not work as well as the other methods.

6. SUMMARY AND FUTURE WORK

In this paper we have explored integration and evaluation of feature selection methods in the context of appearance-based vehicle tracking in geospatial video data. We have developed a test-bed that decouples evaluation of the feature selection module from the rest of the tracking system and we have analyzed four selection methods with varying levels of optimality and computational cost. Feature selection methods combined with the rich feature sets show great promise for various applications. They result in improved quantitative performance, more efficient computational performance, and increased adaptability and flexibility. For the geospatial data processing applications, feature selection becomes even more important, because of the characteristics and the volume of the data processed and the operational challenges. The use of the rich set of features combined with a selection procedure increases adaptability of the overall system to changing operating conditions (i.e. different sensors, altitudes etc.) and variability of the background and foreground appearances under different environment or imaging conditions. The offline feature selection on training data, followed by online processing using the reduced feature set becomes even more critical for real-time processing of large datasets with limited resources such as power-restricted computation on aerial platforms. Our future plans are to incorporate feature fusion methods such as variational-ratio, distracter index, and feature prominence to our evaluation test-bed[18] and to extend the current offline selection process to a semi-online feature selection process similar to the online boosting approaches(MILTrack[44]).

6.1 Acknowledgments

This research was partially supported by U.S. Air Force Research Laboratory (AFRL) under agreement AFRL FA8750-11-C-0091. Approved for public release (case 88ABW-2012-2536). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of AFRL or the U.S. Government. The U.S. Government. is authorized to reproduce and distribute reprints for Government. purposes notwithstanding any copyright notation thereon.

REFERENCES

[1] Palaniappan, K., Bunyak, F., Kumar, P., Ersoy, I., Jaeger, S., Ganguli, K., Haridas, A., Fraser, J., Rao, R., and Seetharaman, G., "Efficient feature extraction and likelihood fusion for vehicle tracking in low frame rate airborne video," in [*13th Int. Conf. Information Fusion*], 1–8 (2010).

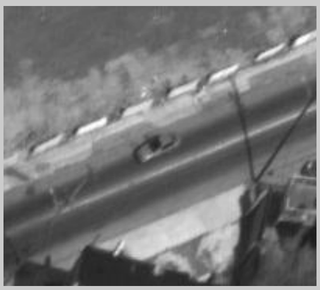
PSS Car#	ROI	Vehicle template	Selected feature subsets	score
1			SFFS={1,2,3,7,8,9,10,11}	1.18
			SFS={6,2,1,5,11}	1.19
			FOCUS={1,2,8,10}	1.14
			RELIEF={9,10,8}	1.22
4			SFFS={8,2,9,5,11,3}	2.19
			SFS={8,2,9,5,11,3}	2.19
			FOCUS={1,2,5,7,8,11}	2.03
			RELIEF={9,8,4,11,3,1,2,10,6,7}	2.42
11			SFFS={1,4,7,8,9,11}	1.01
			SFS={9,11,7,3,8,4,1}	1.09
			FOCUS={1,4,7,8,9,11}	1.01
			RELIEF={2,1,9,8,11}	1.27
14			SFFS={1,2,4,6,8,9,11,7}	1.006
			SFS={1,2,11,4,6,7,8,9}	1.006
			FOCUS={1,2,4,6,8}	1
			RELIEF={4,2,7,11,1}	1.03
17			SFFS={1,2,6,7,3}	1.06
			SFS={1,2,5,7,6,9}	1.06
			FOCUS={1,2,6,7}	1.06
			RELIEF={8,7,1,9,10,11,6,4,3,2}	1.38

Figure 5. Five sample PSS cars, the best selected feature subsets and their associated performance obtained using the four described feature selection methods

Result (SFFS)		
Feature number	Selected features	score
1	{8}	4.13
2	{8,2}	2.33
3	{8,2,7}	2.17
4	{8,2,7,1}	1.98
5	{8,2,7,1,10}	1.92
6	{8,2,7,1,10,9}	1.88
7	{8,2,7,1,10,9,3}	1.94
8	{1,2,3,6,7,8,11,9}	1.96
9	{1,2,3,6,7,8,11,9,10}	2.01
10	{1,2,5,6,7,8,9,10,11,4}	2.10
11	{1,2,3,4,5,6,7,8,9,10,11}	2.092

Result (SFS)		
Feature number	Selected features	score
1	{8}	4.13
2	{8,2}	2.33
3	{8,2,7}	2.17
4	{8,2,7,1}	1.98
5	{8,2,7,1,10}	1.92
6	{8,2,7,1,10,9}	1.88
7	{8,2,7,1,10,9,3}	1.94
8	{8,2,7,1,10,9,3,11}	1.98
9	{8,2,7,1,10,9,3,11,5}	2
10	{8,2,7,1,10,9,3,11,5,6}	2.097
11	{1,2,3,4,5,6,7,8,9,10,11}	2.092

Result (RELIEF)		
Feature number	Selected features	score
1	{8}	4.13
2	{8,5}	3.06
3	{8,5,10}	2.86
4	{8,5,10,11}	2.99
5	{8,5,10,11,7}	2.46
6	{8,5,10,11,7,9}	2.3
7	{8,5,10,11,7,9,6}	2.37
8	{8,5,10,11,7,9,6,3}	2.38
9	{8,5,10,11,7,9,6,3,2}	2.16
10	{8,5,10,11,7,9,6,3,2,1}	2.21
11	{8,5,10,11,7,9,6,3,2,1,4}	2.09

Result (FOCUS)		
Feature number	Selected features	score
1	{8}	4.13
2	{2,8}	2.33
3	{2,7,8}	2.17
4	{1,2,7,8}	1.98
5	{1,2,7,8,10}	1.92
6	{1,2,6,7,8,11}	1.87
7	{1,2,3,6,7,8,11}	1.89
8	{1,2,3,6,7,8,9,11}	1.96
9	{1,2,5,6,7,8,9,10,11}	1.98
10	{1,2,3,4,5,7,8,9,10,11}	2.097
11	{1,2,3,4,5,6,7,8,9,10,11}	2.092

Figure 6. Selected feature subsets, in sorted order, and their performance at different levels using the four described feature selection methods (scores are averaged over all 15 PSS cars on a total of 2290 frames).

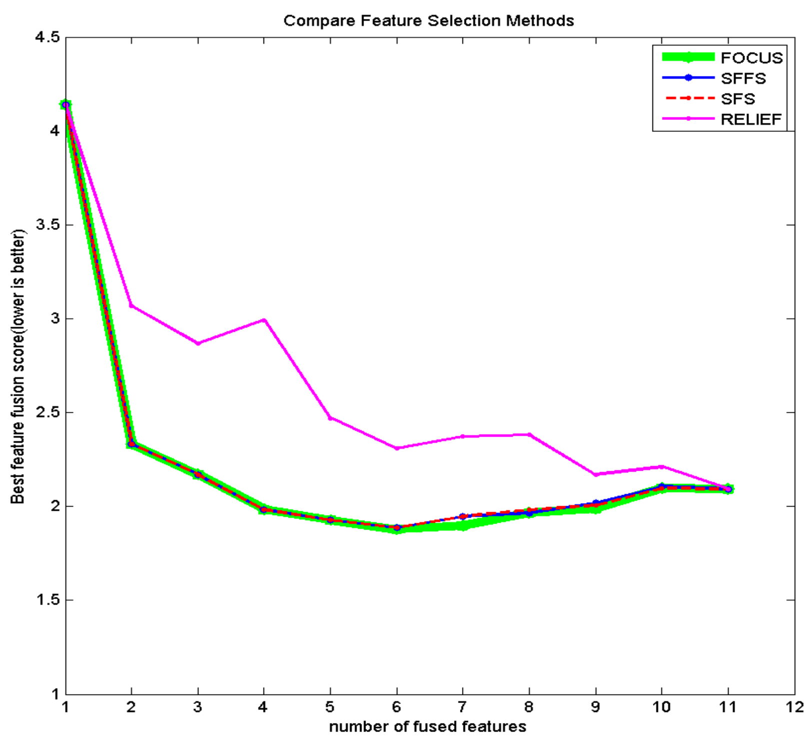


Figure 7. Performance comparison across the four feature selection methods, FOCUS, SFFS, SFS and RELIEF, described in the text with scores averaged over all 15 PSS cars for a total of 2290 frames.

[2] Palaniappan, K., Rao, R., and Seetharaman, G., "Wide-area persistent airborne video: Architecture and challenges," in [*Distributed Video Sensor Networks: Research Challenges and Future Directions*], Banhu, B., Ravishankar, C. V., Roy-Chowdhury, A. K., Aghajan, H., and Terzopoulos, D., eds., ch. 24, 349–371, Springer (2011).

[3] Pelapur, R., Candemir, S., Poostchi, M., Bunyak, F., Wang, R., Seetharaman, G., and Palaniappan, K., "Persistent target tracking using likelihood fusion in wide-area and standard video sequences," in [*15th Int. Conf. Information Fusion*], (2012).

[4] Ersoy, I., Palaniappan, K., Seetharaman, G., and Rao, R., "Interactive tracking for persistent wide-area surveillance," in [*Proc. SPIE Conf. Geospatial InfoFusion II (Defense, Security and Sensing: Sensor Data and Information Exploitation)*], **8396** (2012).

[5] Pelapur, R., Palaniappan, K., and Seetharaman, G., "Robust orientation and appearance adaptation for wide-area large format video object tracking," in [*9th IEEE Int. Conf. Advanced Video and Signal-Based Surveillance*], (2012).

[6] Ersoy, I., Palaniappan, K., and Seetharaman, G., "Visual tracking with robust target localization," in [*IEEE Int. Conf. Image Processing*], (2012).

[7] Jaeger, S., Palaniappan, K., Casas-Delucchi, C., and Cardoso, M., "Classification of cell cycle phases in 3d confocal microscopy using PCNA and chromocenter features," in [*7th Indian Conference on Computer Vision, Graphics and Image Processing*], (2010).

[8] Jaeger, S., Palaniappan, K., Casas-Delucchi, C., and Cardoso, M., "Dual channel colocalization for cell cycle analysis using 3d confocal microscopy," in [*IEEE Int. Conf. Pattern Recognition*], (2010).

[9] Ersoy, I. and Palaniappan, K., "Multi-feature contour evolution for automatic live cell segmentation in time lapse imagery," in [*30th Int. IEEE Engineering in Medicine and Biology Society Conf. (EMBC)*], 371–374 (2008).

[10] Hafiane, A., Palaniappan, K., and Seetharaman, G., "UAV-video registration using block-based features," in [*IEEE Int. Geoscience and Remote Sensing Symposium*], **II**, 1104–1107 (2008).

[11] Palaniappan, K., Ersoy, I., and Nath, S. K., "Moving object segmentation using the flux tensor for biological video microscopy," *Lecture Notes in Computer Science (PCM)* **4810**, 483–493 (2007).

[12] Almuallim, H. and Dietterich, T., "Learning with many irrelevant features," **2**, 547–552 (1991).

[13] Pudil, P., Novovicová, J., and Kittler, J., "Floating search methods in feature selection," *Pattern recognition letters* **15**(11), 1119–1125 (1994).

[14] Devijver, P. and Kittler, J., [*Pattern recognition: A statistical approach*], Prentice/Hall International (1982).

[15] Kira, K. and Rendell, L., "A practical approach to feature selection," in [*International Conference on Machine Learning*], 249–256 (1992).

[16] Blasch, E., Deignan, P., Dockstader, S., Pellechia, M., Palaniappan, K., and Seetharaman, G., "Contemporary concerns in geographical/geospatial information systems (GIS) processing," in [*Proc. IEEE National Aerospace and Electronics Conference (NAECON)*], 183–190 (2011).

[17] Seetharaman, G., Gasperas, G., and Palaniappan, K., "A piecewise affine model for image registration in 3-D motion analysis," in [*IEEE Int. Conf. Image Processing*], 561–564 (2000).

[18] Candemir, S., Palaniappan, K., Bunyak, F., and Seetharaman, G., "Feature fusion using ranking for object tracking in aerial imagery," in [*Proc. SPIE Conf. Geospatial InfoFusion II (Defense, Security and Sensing: Sensor Data and Information Exploitation)*], **8396** (2012).

[19] Candemir, S., Palaniappan, K., Bunyak, F., Seetharaman, G., and Rao, R., "Feature prominence-based weighting scheme for video tracking," in [*8th Indian Conference on Computer Vision, Graphics and Image Processing*], (2012).

[20] Palaniappan, K. and Fraser, J., "Multiresolution tiling for interactive viewing of large datasets," in [*17th Int. AMS Conf. on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography and Hydrology*], 338–342, American Meteorological Society (2001).

[21] Palaniappan, K., Hasler, A., Fraser, J., and Manyin, M., "Network-based visualization using the distributed image spreadsheet (DISS)," in [*17th Int. AMS Conf. on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography and Hydrology*], 399–403 (2001).

[22] Hasler, A., Palaniappan, K., and Chesters, D., "Visualization of multispectral and multisource data using an interactive image spreadsheet (IISS)," in [*8th Int. AMS Conf. on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography and Hydrology*], 85–92 (1992).

[23] Palaniappan, K., Hasler, A., and Manyin, M., "Exploratory analysis of satellite data using the interactive image spreadsheet (IISS) environment," in [*9th Int. AMS Conf. on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography and Hydrology*], 145–152 (1993).

[24] Hasler, A. F., Palaniappan, K., Manyin, M., and Dodge, J., "A high performance interactive image spreadsheet (IISS)," *Computers in Physics* **8**(4), 325–342 (1994).

[25] Haridas, A., Pelapur, R., Fraser, J., Bunyak, F., and Palaniappan, K., "Visualization of automated and manual trajectories in wide-area motion imagery," in [*15th Int. Conf. Information Visualization*], 288–293 (2011).

[26] Porikli, F., "Integral histogram: A fast way to extract histograms in cartesian spaces," in [*IEEE CVPR*], **1**, 829–836 (2005).

[27] Bellens, P., Palaniappan, K., Badia, R. M., Seetharaman, G., and Labarta, J., "Parallel implementation of the integral histogram," *Lecture Notes in Computer Science (ACIVS)* **6915**, 586–598 (2011).

[28] Poostchi, M., Palaniappan, K., Bunyak, F., Becchi, M., and Seetharaman, G., "Efficient GPU implementation of the integral histogram," in [*Lecture Notes in Computer Science (ACCV Workshop on Developer-Centered Computer Vision)*], **7728**(Part I), 266–278 (2012).

[29] Nath, S. and Palaniappan, K., "Adaptive robust structure tensors for orientation estimation and image segmentation," *Advances in Visual Computing* , 445–453 (2005).

[30] Hafiane, A., Seetharaman, G., Palaniappan, K., and Zavidovique, B., "Rotationally invariant hashing of median patterns for texture classification," *Lecture Notes in Computer Science (ICIAR)* **5112**, 619–629 (2008).

[31] Molina, L., Belanche, L., and Nebot, A., "Feature selection algorithms: A survey and experimental evaluation," in [*International Conference on Data Mining*], 306–313 (2002).

[32] Koller, D. and Sahami, M., "Toward optimal feature selection," in [*International Conference on Machine Learning*], 87–95 (1996).

[33] Guyon, I. and Elisseeff, A., "An introduction to variable and feature selection," *Journal of Machine Learning Research* **3**, 1157–1182 (2003).

[34] Caruana, R. and Freitag, D., "Greedy attribute selection," in [*International Conference on Machine Learning*], 28–36 (1994).

[35] Kohavi, R., *Wrappers for performance enhancement and oblivious decision graphs*, PhD thesis, Stanford University (1995).

[36] Kim, Y., Street, W., and Menczer, F., "Feature selection in unsupervised learning via evolutionary search," in [*ACM Knowledge Discovery and Data Mining*], 365–369 (2000).

[37] Das, S., "Filters, wrappers and a boosting-based hybrid for feature selection," in [*International Conference on Machine Learning*], 74–81 (2001).

[38] Xing, E., Jordan, M., Karp, R., et al., "Feature selection for high-dimensional genomic microarray data," in [*International Conference on Machine Learning*], 601–608 (2001).

[39] Liu, H. and Yu, L., "Toward integrating feature selection algorithms for classification and clustering," *IEEE Knowledge and Data Engineering* **17**(4), 491–502 (2005).

[40] Gilad-Bachrach, R., Navot, A., and Tishby, N., "Margin based feature selection–Theory and algorithms," in [*International Conference on Machine Learning*], 43–50, ACM Press (2004).

[41] Jain, A. and Zongker, D., "Feature selection: Evaluation, application, and small sample performance," *IEEE Pattern Analysis and Machine Intelligence* **19**(2), 153–158 (1997).

[42] Julier, S. J., Uhlmann, J. K., Walters, J., Mittu, R., and Palaniappan, K., "The challenge of scalable and distributed fusion of disparate sources of information," in [*SPIE Proc. Multisensor, Multisource Information Fusion: Architectures, Algorithms and Applications*], **6242**(1), Online (2006).

[43] Ververidis, D., "Feature selection using MATLAB," (2010). http://www.mathworks.com/matlabcentral/fileexchange/22970-feature-selection-using-matlab.

[44] Babenko, B., Yang, M., and Belongie, S., "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Analysis and Machine Intelligence* **33**(8), 1619–1631 (2011).