Robust vehicle detection in aerial images based on salient region selection and superpixel classification

Samir Sahli, Pierre-Luc Duval, Yunlong Sheng and Daniel A. Lavigne

Center for Optics, Photonics and Lasers, Laval University, QC, G1K 7P4, Canada;
Defence Research and Development Canada, 2459, Pie-XI Blvd. North, QC G3J 1X5, Canada.

ABSTRACT

For detecting vehicles in large scale aerial images we first used a non-parametric method proposed recently by Rosin to define the regions of interest, where the vehicles appear with dense edges. The saliency map is a sum of distance transforms (DT) of a set of edges maps, which are obtained by a threshold decomposition of the gradient image with a set of thresholds. A binary mask for highlighting the regions of interest is then obtained by a moment-preserving thresholding of the normalized saliency map. Secondly, the regions of interest were over-segmented by the SLIC superpixels proposed recently by Achanta et al. to cluster pixels into the color constancy sub-regions. In the aerial images of 11.2 cm/pixel resolution, the vehicles in general do not exceed 20 x 40 pixels. We introduced a size constraint to guarantee no superpixels exceed the size of a vehicle. The superpixels were then classified to vehicle or non-vehicle by the Support Vector Machine (SVM), in which the Scale Invariant Feature Transform (SIFT) features and the Linear Binary Pattern (LBP) texture features were used. Both features were extracted at two scales with two size patches. The small patches capture local structures and the larger patches include the neighborhood information. Preliminary results show a significant gain in the detection. The vehicles were detected with a dense concentration of the vehicle-class superpixels. Even dark color cars were successfully detected. A validation process will follow to reduce the presence of isolated false alarms in the background.

Keywords: Vehicle detection, aerial image, SVMs, target detection, salient regions, superpixels, over-segmentation, SIFT, Linear Binary Pattern

1. INTRODUCTION

The aerial electro-optical surveillance is widely used for a range of field applications, including: traffic control management, detection of military targets, detection of changes and digital terrain modeling. With multiplication of sensors of different nature and the improvement of their resolutions, large datasets of images have been created, that leads intrinsically to two opposite effects. On the one hand, these datasets represent an increase of valuable data which is a real benefit for airborne surveillance community; on the other hand, this huge amount of information obliges the development and the use of new analysis techniques to threat the data. Amid the applications that have received great attention over the last years in the remote sensing community, we can notice the detection of vehicle in large scale imagery acquired by high resolution sensors. Different strategies have been developed for detecting car in this context. For instance, vehicles can be considered as explicit 2D instances and be detected with edge masks by extracting the four sides of rectangular boundary [1], or also modeled with rectangles of different sizes [2].

In this paper, we present a set of algorithms for vehicle detection in large scale aerial images using the geometric and radiometric feature extraction techniques. Vehicles, which we desired to detect, interfere with a complex, real and uncontrolled environment and produce occlusions and shadow areas within the scene. At the 11.2 cm/pixel resolution, a conventional passenger car appears within an area of 20 x 40 pixels size. Therefore, only large subcomponent of vehicles such as the roof, the trunk and the windshield can be resolved. Plus, these image features may seem distorted by the noise, specular reflection, low color contrast level and viewpoint variation. An additional challenge in the airborne detection task is to cope with the wide area coverage required for keeping a global awareness of the scene, and the maintaining of the detection capability of small targets in large scale images. These both aspects should be considered to avoid an unreasonable search time in the entire scene but also to guarantee that no vehicle have been lost in the process.

*sheng@phy.ulaval.ca; phone 1 418 656-2131, ext. 3908; fax 1 418 656-2623
Based on these observations, we first define regions of interest, where vehicles may appear. The avowed aim is to focus efforts on limited areas but also reduce the prospection time for detecting vehicles. Then, we over-segment the regions of interest to obtain clusters of pixels with same color properties. This pre-processing permits a sub-regions analysis rather than working separately on each pixel in the image. This approach is less expensive and more effective in the context of airborne surveillance. Finally, local and global radiometric features are extracted from each superpixel and encoded into features vectors as signatures. The generalization capability of the Support Vector Machine (SVM) is then used to classify these signatures as vehicle or non-vehicle.

2. METHODOLOGY

2.1. Salient region selection

In the context of vehicle detection in the large scale aerial imagery, it is essential to select regions of interest to reduce the computation time. The region selection differs from the traditional segmentation techniques [3, 4] because segmentation has for objective to divide the image to obtain regions semantically coherent. The segmentation is subjective and depends on the objects nature that we desire to detect.

For its parts, the salient region determination consists on binary separation of the image in order to highlight regions which catch our visual attention. In this goal, a salient map is estimated by extracting structuring elements of the images, such as edges, corners, dominant colors. By mapping a salient map, we attempt to assign a saliency value for each pixel which reflects the confidence that we carry into the given pixel to belong to a relevant part of the visual field or to an object that we desire to detect. Several algorithms have been proposed over the years in order to cope with this challenging problem. Koch and Ullman [5] were among the first to propose a solution for salient map computation. This method has been further developed by additional work of Itti [6] and Walther [7]. In summary, the salient map is determined by the dissimilarity that exists from one pixel and its neighborhood in terms of geometric and radiometric features such as, color, intensity or orientation. Then, all the collected information are summed pixel by pixel and normalized to obtain the salience map. In recent years, many other approaches have been developed. For instance, Hou and Zhang [8] propose a technique independent of the features, categories or other forms from the prior knowledge of the object. A log-spectrum analysis is done to extract the spectral residual of the image in the spectral domain. This technique is based on the observation that the average Fourier spectrum of the ensemble of natural images obeys a 1/f distribution. The singularity in the smooth distribution betrays the presence of anomalous regions that may contain relevant object. Then, the salience map is simply computed by estimating the Inverse Fourier Transform of the spectral residual matrix. In the specific context of broad area overhead or satellite images, Li and Itti [9] introduce a salient map in terms of a set of biologically-inspired low-level visual features. The method combines a large number of salient informations such as intensity, orientation, surprise, variance, entropy, spatial correlation but also X, L or T junction in order to create a salient map.

In any case, the underlying principle for the design of a salient map is first, the determination of the characteristics which best represent the situation to highlight or the object to detect in the image. Then, in light of this knowledge a tool can be created. Vehicles are a man made structure which is composed by sub-component such as the windshield, the windows or the roof. These different parts contain many edges which are close to each other and belong to a limited area. If we consider edges pixels relatively to the surface covered by them, we can safely formulate the following assumption: vehicles appear where regions have high edges density. In the literature, the edge density is usually evaluated by using a Parzen estimator [10]. Nevertheless, this method requires an appropriate kernel size that depends on the image nature. An alternative solution is to propagate the edge information through the image by applying distance transform (DT) [11] from an edge map. Pixels close to edges have low values, whereas pixels far from them have higher distance value. Rosin [12] exploits this idea and defines a simple and a non-parametric method for salient map determination based on edges density calculation. In the rest of this paper we adopt the salient regions method proposed by Rosin for detecting regions where the vehicles can appear. The method can be divided in two steps: 1) salient map calculation and 2) thresholding for highlighting salient regions. The algorithm is detailed as follows:

The image is firstly converted from sRGB to gray level image and the gradient magnitude $E$ is evaluated and saved in a matrix with the same size that the original image. A collection of equally spaced thresholds is set to obtain a list of $p$ different thresholds (in practice $p$ is equal to 64). Successively, each listed value is used to threshold the gradient magnitude $E$ and as result an edge map $Ei$ is obtained. Then, the information of pixel edges map is propagated through the image by applying the distance transform DT. Each pixel of the distance map $DT(Ei)$ contributes for $1/p$ of the
resulting salient map since all computed \( DT(E_i) \) distance map are accumulated in a matrix, as a final step. We can notice that salient map calculation is free of any parameters.

An example of salient map computed from a large scale aerial image is illustrated in the Fig.1b). In the image, the edge density is proportional to the gray value. The lower the pixel value is, the higher the edges density is. By comparing both images of the first row in the Fig.1, we can see pixels belonging to car regions are darker than the rest of the scene. This observation confirms our assumption that vehicles appear in regions with high edges density.

The next step of the algorithm consists of the determination of the salient regions, which the number, the respective size and the shapes are \textit{a priori} not known. In this manner, Rosin uses Tsai’s moment preserving algorithm [13] to automatically determine threshold value and provide a binary mask that highlights the salient regions. The Fig.1c) illustrated the result obtained with Rosin’s algorithm for an ordinary scene of a parking lot obtained in nadir view.

Fig 1. a) large scale image of parking lot in nadir view; b) salient map obtained with Rosin method [12] c) overlay of a grayscale image with mask obtained by the thresholding of the salient map with Tsai’s moment preserving algorithm[13].

2.2. Over-segmentation and superpixels

The over-segmentation is a pre-processing that creates compact and contiguous regions composed by pixels which have similar properties. The interest of superpixels resides on the manipulation of sub-regions rather than isolated pixels; and leads to a significant reduction of image complexity. The superpixels are defined with the respect of one principle. \textit{The superpixel area should never been greater than an image sub-region for which a meaning could be associated in the image.} For instance, we can see in the Fig.2 an isolated car on the road obtained in a nadir view. Pixels that composed its roof are clustered together and several superpixels appear. However, none of the obtained sub-regions can be related to a specific meaning from the image; they only represent clusters of pixels that share the same color. This constraint avoids under-segmentation error through the pixel grouping process. The under-segmentation can be assimilated to a kind of “bleeding” effect dues to the presence of large superpixels that overlap structures without respecting inherent details in the image.

In order to over-segment adequately the image, a particular attention has to be taken during the splitting process to preserve the inner edges of the objects but also those belonging to the objects boundaries. Consequently, it is clear that the superpixels size is a crucial parameter that should be properly chosen in light of the pixel resolution and the importance devoted to the details in the image. In the Fig.2, the superpixels size is approximately equals to 45 pixels. This choice provides a contours map that coincides pretty well with boundaries of the shadow and vehicle components such as windshield, lateral windows and roof. This superpixels size seems to be a good comprise for capturing vehicle details for the given resolution. Thus, for the rest of the experiment, we over-segment images to obtain superpixels with surface area roughly equals to 45 pixels. By introducing this \textit{size constraint}, we guarantee that no superpixels exceed the size of a vehicle and limit the under-segmentation error to occur.

Superpixels are becoming increasingly popular in computer vision applications. Many implementations have been developed in the literature [14-17]. For our aerial surveillance application, we retain as a tool, the Simple Linear Iterative Clustering (SLIC) superpixels, a method proposed recently by Achanta \textit{et al.} [18]. The SLIC superpixel method meets all requirements we need such as, low computational cost, high quality segmentation that uses perceptual CIELAB color space and the compactness warranty for the all superpixels. The SLIC method clusters pixels based on their color \textit{similarity} and \textit{proximity} in the image plane and needs, at the onset, the number of desired superpixels \( K \). For an image
with \( N \) pixels, the approximate size of each superpixel is therefore \( S^2 = N / K \) pixels. A description of SLIC superpixel algorithm is introduced below.

**Fig 2.** a) A close up view of typical vehicle at 11.2 cm/pixel; b) result of the over-segmentation by SLIC. Each superpixel does not exceed an area of 45 pixels.

After converting an assuming sRGB input image to CIELAB color space, an initialization step is set by sampling on the image a grid of \( K \) regularly spaced cluster centers \( C_k \), with \( k \in [1,K] \). Each center corresponds to the initial position of a superpixels centroid. Since the spatial extent of any superpixels is approximately \( S^2 \), we can safely assume that pixels, associated with this cluster center, lie within a \( 2S \times 2S \) area. The keystone of the SLIC algorithm resides on the manner how the pixels are respectively assigned to the \( K \) centers. In this aim, a weighted distance measure \( D_s \) is introduced to enforce color similarity as well as proximity:

\[
D_s = D_{lab} + m \frac{d_{xy}}{S},
\]

A five-dimensional space is created with \([lab]\) color vector for the CIELAB color space and \([x,y]\) for pixel coordinates in the image plane. The spatial distance \( d_{xy} \) is normalized by a factor \( S \), which represents the interval distance existing between two neighbor centers. The goal is to introduce an upper bound for limiting the maximum permitted value of \( d_{xy} \) and reducing the \( d_{xy} \) magnitude to match with that of the colorimetric distance \( d_{lab} \). The \( k \) corresponds to the index of the \( K \) possible centers and the index \( i \) identify pixels that lie within the area around a given superpixel center \( k \).

The \( m \) value is introduced to modify the respective contribution of spatial and colorimetric terms in the distance measure \( D_s \). During the clustering process, the smaller the \( m \) value is, the greater the emphasis is put on the color. In our experiment, we use \( m=5 \) in order to encourage a higher color sensibility and solve boundaries discrimination problem that occurs in presence of color dark cars. Hence, with the conjunction of small \( m \) value and the introduction of a constraint on the size of superpixels adequately chosen, it is possible to discriminate and discern properly windshields and vehicle shadows from the other parts of the vehicle, as illustrated in Fig.2. The next step of the algorithm is to associate each pixel in the image with the nearest cluster center \( C_k \) according to the weighted measure distance \( D_s \). Then, a new center is computed as the average of \([labxy]\) vector of all pixels belonging to the same cluster. This can be compared to a migration of centers to reach new positions in the \( xy \) image plane. The assignment process for pixels is iteratively repeated until centers position of all clusters converges to stable location.

The following subsections describe how the local and global features are extracted from superpixels and formatted as a signature in order to be used by Support Vector Machine for classification.

### 2.3. Local and global feature extraction

Once the regions of interest are over-segmented by introducing the size constraint, a set of color constant sub-regions is obtained. Each of them covers approximately the same area in the image but their respective shape differs
depending on their locations. This reflects a particular relation that exists between the superpixels boundaries and their internal regions. Due to the iterative assignment process of SLIC, the superpixels are able to adapt their shapes with color structures present in the image. In summary, centers migration plays an important role for reaching an optimal over-segmentation.

Since the center determination is the result of colorimetric and proximity optimization and these centers average the colorimetric information of the pixels into respective superpixel, we propose to consider the superpixel centroid as a keypoint, like other feature or region detector recorded in the literature [19]. By modifying the desired superpixels size, we can indirectly control the location of centers but also the number of keypoints that appear on each vehicle.

As proposed in [20, 21], we will encode information around each keypoint as a signature. However, our approach differs in that we capture not only local structure but also global information by using two different size patches; both are designed in sight of the vehicle size at the given resolution (all further details are given in the sub-section). The small patch extracts fine details of vehicles as windshield, roof; whereas the large patch catches neighborhood information present in the surrounding of the vehicle. Information related to each patch will be store respectively in a feature vector that will be used as inputs for classify superpixels as vehicle or non-vehicle class.

Feature vector consists on a concatenation of texture and radiometric information computed with Linear Binary Pattern (LBP) [22] and Scale Invariant Feature Transform (SIFT) [23] descriptor, respectively. The LBP is a statistical texture analysis evaluated from pixels values distributed through a gray level image. The texture is encoded as a pattern and describes the relationship that exists between the gray value $g_c$ of the center pixel $c$ and its local neighborhood pixels $p$ with respective $g_p$ gray values, located at a distance $R$ from the pixel $c$:

$$LBP_{p,R} = \sum_{p=0}^{P-1} s(x)2^p \quad x = g_p - g_c \quad \text{where,} \quad s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$  \hspace{1cm} (2)

In the neighbor of the pixel $c$, $P$ pixels are ordered in a counter-wise list and for each of them a value $2^p$ is associated. For all $P$ pixels in the neighborhood of pixel $c$, the sign of the difference $s(x) = g_p - g_c$ is evaluated. When sign is positive, a value $2^p$ corresponding to the pixel at the position $p$ is added to a variable. This process is repeated for each $P$ pixels in the neighborhood to obtain an accumulated value which is associated to the pixel $c$. This value corresponds to a certain texture pattern and it is named a LBP code. By considering only the sign of the difference, the LBP pattern becomes invariant against any monotonic transformation of the gray scale, which is a valuable property. Furthermore, there are $2^p$ different patterns corresponding to the exhaustive list of patterns that can be created with $P$ pixel at a distance $R$ from a center pixel $c$.

When the image is rotated, the pixels in the neighbor move along the perimeter $R$ around the center pixel $c$. Therefore, the value associated to the new pixel location is changed but the pattern stays the same. In order to remove the effect of rotation and guarantee the existence of a unique identifier for each pattern, a Rotation Invariant Local Binary Pattern is introduced and defined as:

$$LBP_{p,R}^{ri} = \min \{ ROR(LBP_{p,R} \cdot i) \quad \mid \quad i = 0,1,\ldots,P-1 \}$$  \hspace{1cm} (3)

where $ROR(x,i)$ performs a circular bit-wise right shift on $P$-bit number $x$ , $i$ time. In this way, for given value $R=1$ and $P=8$, there are 36 unique rotation invariant binary patterns due to the circularly symmetric consideration. A comparative study, of different versions of LBP textures measures [24] has shown that $LBP_{p,R}^{ri}$ does not provide very good texture discrimination. Consequently, we would rather use the Rotation Invariant uniform Patterns [22], and improved version of $LBP_{p,R}^{ri}$ that also considers uniform patterns and defined as follows:

$$LBP_{p,R}^{ui} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c) & \text{if } U(LBP_{p,R}) \leq 2 \\ P+1 & \text{otherwise} \end{cases}$$  \hspace{1cm} (4)

where

$$U(LBP_{p,R}) = \left| \sum_{p=0}^{P-1} (g_{p-1} - g_c) - (g_0 - g_c) \right|$$  \hspace{1cm} (5)

The $LBP_{p,R}^{ui}$ differs from $LBP_{p,R}$ by the introduction of uniformity measure $U$ shown in (5), which corresponds to the number of spatial transitions (bitwise 0/1 changes) in the “pattern”. By setting $P=8$ and $R=1$ or $P=8$ and $R=2$, a set of 10
different unique patterns are created. All the LBP patterns estimated from pixels region are then collected in a normalized histogram of 10 bins (one bin per unique pattern), to be used as signature of the texture present in the region. In our application, the \( LBP_{\gamma} \) is used with \( P=8, R=1 \) and \( P=8 \) and \( R=2 \). This choice is motivated by necessary improvement of discrimination performance for the texture descriptor. On the one hand, \( LBP_{\gamma} \) with a radius \( R=1 \) captures closest information from the vehicle and, on the other hand those with \( R=2 \) catches farther information without exceeding the limited area occupied by the vehicle. In fact, as we will see further in this paper, the texture seems adequate to distinguish structures of dark cars.

The SIFT descriptor was computed for each keypoint i.e. superpixels centroid, so as for each of them a \( D \times D \) patch region oriented to the dominant direction of gradient was used in order to compute a set of \( 4 \times 4 \) histograms of gradient orientation. This histogram consists of a collection of gradient magnitude in each of 16 quadrants and according 8 different orientations. Consequently, each keypoint is described by \( 8 \times 4 \times 4 = 128 \) SIFT features vector. With knowledge about the conventional passenger car dimension – \( 20 \times 40 \) pixels at 11.2 cm/pixel – we propose to extract local and global information from each superpixel by considering two patches of different sizes. The smaller one has a \( 14 \times 14 \) pixel size that covers area smaller than typical vehicle dimension. In this way, it is possible to describe subcomponent parts of vehicles (such as the trunk, hood, and windshield). The largest patch has a \( 28 \times 28 \) pixel size which allows including neighborhood information from environment close to the vehicle.

In summary, texture and radiometric features are extracted from each superpixel and concatenated to obtain a signature of 148 feature elements. Each feature vector contain a \( 10 + 10 \) texture elements plus a \( 128 \) SIFT element. For each superpixel a local and a global feature vectors are associated, both differs from the SIFT patch size but have the same dimension, i.e. 148 feature elements.

### 2.4. Training with Support Vector Machine (SVMs)

Support Vector Machines is a supervised learning algorithm proposed by Vapnik [25] for classification problem. The algorithm solves an optimization problem that consists on the determination of a set of hyperplanes in high dimensional feature space that classifies adequately a set of feature data in two or more pre-determined classes. In contrary to Neuronal Network algorithm [26], the SVM does not require a thorough knowledge about architecture design. Recent improvements in the SVM include the soft margin [27] and the regression problem solving [28]. The cornerstone of the algorithm is that it seeks not only to find the hyperplanes that adequately separate the data belonging to different classes, but it also tries to determine the hyperplanes that provide the largest possible margin. In prediction phase, that confers to the SVM a high capacity of generalization with the introduction of certain flexibility due to the margin.

The SVM needs to be trained with a collection of representative feature data i.e. learning samples for which labels are associated (+1 and −1 for classification problem and real value for regression). For a learning set \( A = \{(x_i, y_i) \}_{i=1}^{k} \) where \( x_i \in \mathbb{R}^n \) and \( y_i \in \{-1, +1\} \), the algorithm maps the \( x_i \) vectors in subspace \( \mathbb{H} \) using a linear function \( \phi : \mathbb{R}^n \rightarrow \mathbb{H} \) in order to determine the optimal hyperplan \( (w, b) \) separating the two classes of \( \mathbb{H} \). The new class \( y \) of the sample \( x \) is then defined by \( y = sign(w \cdot \phi(x) + b) \). The hyperplan is considered optimal if it maximizes the distance between its closest neighbor samples.

The superpixels signatures will be classified by training the SVM with a collection of superpixels features that correspond to vehicle region (\( y = +1 \)) and non-vehicle region (\( y = -1 \)). The respective signature will be map into a feature-space in which they can be separated by a linear hyperplan, using the Euclidean distance between feature vectors \( x_i \) and \( x_j \), by means of the radial basis function kernel:

\[
\psi(x_i, x_j) = \phi(x_i) \phi(x_j) = \exp(-\gamma \|x_i - x_j\|^2) \tag{6}
\]

where \( \gamma \) is hyper-parameter. The corresponding dual problem can be represented by the decision function:

\[
f(x) = \sum_{\alpha_i \in \mathcal{Z}} \alpha_i y_i \psi(x_i, x) + b \tag{7}
\]

where the Gaussian kernel is defined by \( \psi(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \) and \( \{\alpha_i\} \) and \( b \) corresponds to the optimal solution of the dual problem. With the soft margin model [27], the SVM allows the mislabeled examples during the training phase.
Therefore, the dual problem is reformulated as a constrained functional:

$$\min_{\mathbf{w}, \xi} \left( \frac{1}{2} \mathbf{w}' \mathbf{w} + C \sum_i \xi_i \right) \text{ subject to } y_i (\mathbf{w}' \phi(x_i) + \mathbf{b}) \geq 1 - \xi_i$$

where $\xi_i > 0$ is the slack variable which measures the degree of misclassification, $C > 0$ is the penalty parameter of classification errors, $\mathbf{w}$ is the support vector normal to the hyperplane, $\mathbf{b}$ is the offset to the hyperplane to the origin. These hyperparameters $\xi_i$ and $C$ are data-dependent and have to be determined by a cross-validation method [29] on the training set.

3. EXPERIMENTS

3.1. Salient region selection and superpixel

A set of aerial image of 1000 x 1000 pixels with resolution of 11.2cm/pixel was used as relevant dataset to test the proposed algorithms. Although the spatial resolution of the dataset can be considered as relatively high but the fact remains that it is still a too low resolution for extracting vehicles at this resolution since such typical instances appear within a rectangle of 20 x 40 pixels size (c.f. Fig.1a). Therefore, only large subcomponent parts of vehicles (such as the trunk, hood, windshield, and associated shadow) can be distinguished at the given resolution. Additionally, image related features to be extracted from the vehicles are distorted by induced noise, specular reflection, viewpoint variation, and low color contrast level.

Our vehicle detection algorithm relies on 5 steps: 1) determine interest region with Rosin’s method for keeping only regions with dense edges; 2) over-segment the image by using the SLIC superpixels and a size constraint to avoid under-segmentation errors and to reach an optimal over-segmentation 3) fuse the first two steps to obtain a collection of superpixels belonging to the interest regions; 4) extract global and local information with LBP texture and SIFT radiometric descriptor to create two different signatures per superpixel; 5) use this set of signatures to train a SVM and create a predictive model.

For training the SVM, three images of 1000 x 1000 pixels goes through steps 1) to 4). This training set is used to create a model which allows in the predicting phase to classify new examples that the SVM has never seen before. However, firstly the hyperparameters $\xi_i$ and $C$ in Eq.8 have to be defined since SVM is data-dependant algorithm. We used a 10-fold cross-validation technique [29]. By dividing the training set into 10 equal-sized sub-sets and using alternatively one set for prediction and leave the other for training, we can scan all possible hyperparameters and determine which couple $(\xi_i, C)$ represents the best choice for creating a SVM model with the highest prediction capability. For our experiment and the available data set, we determine that the best couple is $(\xi_i, C) = (0.0078, 512)$.

We ought to notice that for training set we only use the local descriptor which correspond to $14 \times 14$ SIFT patch size. This choice is motivated by two reasons. The first one resides on the fact, SVM algorithm takes time to converge due to the large number of signature. To reduce the time of computation, we can use the primal formulation of optimizing
problem. In this way, time is saved for sampling data with Gaussian kernel function. However, nothing guarantees that hyperplanes with large margin will be found since we lost the flexibility introduced by the use of the Gaussian kernel function. In addition, only one SVM model will be used in the prediction phase that reduces the number of Support Vectors to be saved. The concatenation of the both local and global signature is not reasonable since the texture feature will be drowned in the mass of information.

The second reason is related to the nature of the 148 feature descriptor. In fact, the global and local descriptor differ only about their SIFT signature which is computed for two different size patches. A comparative study held by [30] has shown that the SIFT descriptor is relatively invariant against a scale factor change, more precisely until a factor of 2.2. Therefore, we can safely consider that even if the local and global descriptor does not cover the same surface area and consequently they do not dedicated to the same structure in the image, their respective signature should be roughly invariant.

At the end of the training stage, a set of Support Vectors is obtained that solves the optimization problem. In order to measure the efficiency of the SVM model created, we use two different statistical measures: **Accuracy** (i.e. \( \frac{\text{True Positives} + \text{True Negatives}}{\text{the total of predicted data}} \)) and **Sensitivity** (i.e. \( \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \)). In the other words, the accuracy reflects how the vehicle feature and background are well predicted, while the sensitivity relates the ratio of well classified feature as vehicle, versus the total number of feature that appear on the vehicle.

However, we have to precise that during the cross-validation, we used an alternative measure for estimating the hyperparameters. In fact sensitivity is not a good criterion for cross validation. The training dataset that has been built for this experiment contains more negative example than positive examples. This choice is motivated by the desire to catch the largest possible number of different backgrounds present in the image. For unbalanced training set (le number of Positive examples is smaller than those associated to Negative) there exists a risk that the SVM predicts all training examples in one class that involves a 100% of sensitivity. An alternative tool is the **Balanced Accuracy** defined as \( \text{BAC} = \frac{\text{Sensitivity} + \text{Specificity}}{2} \); where \( \text{Specificity} = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}} \). At the end of the the cross-validation processing, we obtain respectively a \( \text{BAC} = 71.98\% \), \( \text{Sensitivity} = 50.47\% \) and \( \text{Accuracy} = 86.92\% \).

The support vector model is now created and will be serve to predict new examples belonging to testing set as presented in the next sub-session.

Before going to the prediction sub-section, we are going to analysis the optimization conditions that guarantee a higher capability of prediction in regard to SVM architecture and the theory of learning algorithm. Two kinds of error exist, the empiric error and the generalization error. The SVM algorithm like many other learning algorithm tries to minimize the empiric error on a training set from which a predict model is empirically created. Whereas, by using the SVM for predicting new data, we desire to have a very low error of generalization (errors related to new data that the SVM has never seen before). An important result of the learning theories is that empiric risk decreases progressively with the complexity of the data. The complexity of the data is inherent to the complexity with which the hyperplanes have to be design in order to match perfectly with the training data. The risk is the hyperplan minimizes the error for training data set but lost its generalization capability for prediction on testing set. This effect is known to **over-fitting error** in the literature. In general, the experiment shows that a simple classifier which make few number of mistakes can have better capability of prediction that a more complex classifier which have no mislabeled example on the training set. By training a SVM, we have to keep in mind that there exist a tradeoff between the complexity and the empiric risk. This resumes the most important principle of induction for learning theories.

### 3.2. Outputs of the SVMs

The testing set is composed by a large number of 1000x1000 pixels images with a resolution of 11.2 cm/pixels. The image contains a wide range of vehicle types that potentially can occur within typical aerial imagery. The aerial imagery describes complex and uncontrolled environment as we can find in a typical and real urban scenario.

In the Fig.4a), the blue points are associated to superpixels centroids which the signature is predicted as belonging to non-vehicle class. The red points are associated to vehicle class. As input, two signatures – local and global – are provided to the SVM. One corresponds to the local structure signature and the other to the global structure for each superpixel. As an output of each superpixel, the SVM provides two predictive values, one for each kind of signature. Finally, we combine these labels by using the **OR logical operator**. This fusion lead to the fact that: **as soon as one of the**
both signatures has been predicted as vehicle class, the considered superpixel is assigned to the vehicle class. The goal is to be sure that we do not lose any of vehicles in the image.

This fusion increases the efficiency of final detection since the local properties become as importance as the global properties in the decision process, and vice versa. In this way, we guarantee that as soon as the SVM predicts a label +1, whatever the nature of the signature (i.e. global or local), the corresponding superpixel is directly associated to the vehicle-class. Finally, we reach an accuracy value and a sensitivity value respectively equal to 90.91% and 62.63%, for the total collection of testing image.

If we focus our intention on the bottom left image of the Fig.4a), we can see superpixels belonging to vehicle are detected even if only few pixels of the vehicle appear in the image. This observation suggests that the proposed method is robust against partial occlusions for detection of the vehicles.

![Fig 4](image_url)

Fig 4: a) colored superpixel centroids illustrate the class prediction of the SVM (Blue for non-vehicle and red for vehicle). There is a dense concentration of centroids predicted as vehicle class; b) in green superpixels which the centroids appear in red in left picture; c) confidence map of pixels which potentially belong to a rectangle center with 20 x 40 pixels size.

In addition, dark car are also detected even if the contrast is pretty low. This result is due to the use of both texture and global feature. The descriptor SIFT is based on histogram calculation of gradient and in situation of low contrast is as for dark color vehicle, the SIFT histogram is flat and no information can be extracted. However, as explained in the sub-section 2.3, the LBP texture descriptor has a prior property: it is invariant against any monotonic transformation of the gray scale. In addition by using global descriptor, structural information in the neighborhood of vehicle can be successfully extracted. In fact, since the radiometric information provided by SIFT descriptor for 28x28 patch size is not limited to local dark sub-component of the vehicle, the SIFT signature can extract information which is extended beyond the boundaries of the dark color vehicle. The next sub-section deals potential validation technique that is further explore in the future work.

### 3.3. Validation

From results of an informal psychological test, Zhao and Nevatia [31] classify the main four factors that help humans for vehicle detection. As visual appearance elements, the rectangular shape is the most important cue. The frontal and/or rear windows, car shadow when it exists and the immediate environment surrounding the car are also considered as relevant factors. Based on these cues, we propose to use Hough Transform (HT) adapted to rectangle detection [32]. The algorithm can be used to detect multiple approximately rectangular shapes with arbitrary orientations but fixed scales.

To proceed, we first applied edge detector on the sub-regions, defined in green in the Fig.4b), which is previously dilated with a morphological operator. With the binary edge map obtained, we looking for positions of potential rectangle centers which the length and the width are equal roughly to 40 x 20 with a tolerance of +/-5 pixels per dimension. The HT algorithm relies on the observation that: an edge pixel with a specific orientation can belong to the long edges or to the shortest edges of the rectangles. In the both case, if the rectangle dimension is known, an approximate position of rectangle center can be evaluated. For one edge pixel, the rectangle center coincides with four line segments parallel. These regions where the center may appear are located at a distance of (+/-)width/2 in the case
where the pixel edge belong to the longest rectangle side and at a distance of (+/-)length/2 if the pixel edge belong to shortest side of the rectangle. According this observation, each edge pixel in the dilated green region is used as evidences in a 2-D accumulator array to build a confidence map of the presence of a rectangle at a given pixel in the image. Figure 4c) represents the confidence map that a rectangle with a predetermined dimension occurs in regions delimited by green color in Fig.4b).

In this way, we associated a low weight to the regions detected in Fig.4b) which do not contain rectangles with the chosen size. These regions could be considered as false alarms. This validation technique is a trivial, nevertheless other existing method can be used to reduce sporadic false alarms. For instance, we can use infra-red image, or shadow detection for capture the surrounding shadow that appear close to the car, cross-correlation is also an efficient method but it is not robust to rotation or partial occlusion. Digital Terrain Model (DTM) may erase false detections that appear on buildings. In a future work, we further focus on this problem of validation.

4. CONCLUSION

We have successfully applied a non parametric method of Rosin to automatically define the regions of interest with the promises to find objects. This step is important for detection of objects in aerial imagery. We extract the texture and SIFT features at the centroid of every SLIC superpixels instead of the feature points. With implementation of the SVM training and testing based of the extracted features we obtained positive alert on all the vehicles to detect, even on the dark-colored vehicles, which are among the most difficult objects to detect. The approach is promising and can be easily followed by a validation process to achieve high performance detections.

ACKNOWLEDGEMENTS

This work has been supported by grants from Natural Sciences and Engineering Research Council of Canada, Fonds Quebecois de Recherche sur la Nature et les Technologies and Defence Research and Development Canada - Valcartier.

REFERENCES