Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Detection of engineering vehicles in high-resolution monitoring images*

Xun LIU[†1], Yin ZHANG[†‡1], San-yuan ZHANG[†1], Ying WANG[†1], Zhong-yan LIANG[1], Xiu-zi YE[2]

([1]*College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China*)

([2]*College of Mathematics and Information Science, Wenzhou University, Wenzhou 325035, China*)

[†]E-mail: star.liuxun@gmail.com; yinzh@zju.edu.cn; syzhang@zju.edu.cn; maggiewang0427@gmail.com

Received Jan. 20, 2015; Revision accepted Mar. 23, 2015; Crosschecked Apr. 9, 2015

Abstract: This paper presents a novel formulation for detecting objects with articulated rigid bodies from high-resolution monitoring images, particularly engineering vehicles. There are many pixels in high-resolution monitoring images, and most of them represent the background. Our method first detects object patches from monitoring images using a coarse detection process. In this phase, we build a descriptor based on histograms of oriented gradient, which contain color frequency information. Then we use a linear support vector machine to rapidly detect many image patches that may contain object parts, with a low false negative rate and a high false positive rate. In the second phase, we apply a refinement classification to determine the patches that actually contain objects. In this stage, we increase the size of the image patches so that they include the complete object using models of the object parts. Then an accelerated and improved salient mask is used to improve the performance of the dense scale-invariant feature transform descriptor. The detection process returns the absolute position of positive objects in the original images. We have applied our methods to three datasets to demonstrate their effectiveness.

Key words: Object detection, Histogram of oriented gradient (HOG), Dense scale-invariant feature transform (dense SIFT), Saliency, Part models, Engineering vehicles

doi:10.1631/FITEE.1500026 **Document code:** A **CLC number:** TP391.41

1 Introduction

In this paper, we present a new formulation for detecting objects with articulated rigid bodies in high-resolution images. Our method is very accurate and fast. There have been significant developments in image detection over the last decade. Most methods scan images to find interesting objects. More precise results can be achieved with more computationally complex procedures, but they are more time consuming, such as Dalal and Triggs (2005), Liu *et al.* (2008), and Rahtu *et al.* (2010). Most image detection methods cannot be directly applied to high-resolution images because they require expensive operations. However, high-resolution cameras are being increasingly applied to monitoring tasks. Our method rapidly detects all objects that are similar to a goal object using the histogram of oriented gradient (HOG) features (Dalal and Triggs, 2005) with color frequencies. Then it produces an accurate classification to determine the goal object, by applying a dense scale-invariant feature transform (SIFT) descriptor (Liu *et al.*, 2008) with a saliency mask.

Dalal and Triggs (2005) applied the HOG descriptor to human detection, using a method that is strongly based on the popular SIFT algorithm from Lowe (2004). Since then, many researchers have applied it to intelligent image processing, in

‡ Corresponding author

* Project supported by the China Knowledge Centre for Engineering Sciences and Technology (No. CKCEST-2014-1-2), the Zhejiang Provincial Natural Science Foundation of China (No. LY14F020027), and the National Natural Science Foundation of China (No. 61272304)

ORCID: Xun LIU, http://orcid.org/0000-0002-3045-2943

areas such as aided driving (Zaklouta and Stanciulescu, 2014) and pedestrian detection (Li *et al.*, 2013) and recognition (Déniz *et al.*, 2011; Kobayashi, 2013). Kobayashi (2013) constructed features using HOG probability density functions (PDFs), and their method performed well. Local objects can be accurately characterized by the distribution of edge directions or local intensity gradients based on their appearances and shapes. The HOG extracts the gradient direction histogram of the local area and considers it a feature of the local image. Then they optimized the features and used a classifier such as a support vector machine (SVM) to detect the object. Avidan (2006)'s feature vector contains the RGB values of pixels, which are used to improve weak classifiers. Ott and Everingham (2009) computed color-HOG descriptors on a soft segmentation of pixels into foreground and background. In this paper, we concatenate the selected frequency of the RGB channels to improve the HOG descriptor of Dalal and Triggs (2005) using simple calculations.

Lowe (2004) proposed the SIFT descriptor for constructing image features. It is scale, rotation, and affine invariant and is not sensitive to illumination. Since it was first introduced, many SIFT variations have been proposed. Ke and Sukthankar (2004) used principal component analysis (PCA) for the normalized gradient patch and found that PCA descriptors are more robust to illumination and rotation than SIFT descriptors. Bay *et al.* (2008) presented a new detector and descriptor called SURF (speeded-up robust features). The extraction and comparison of features are more rapid than conventional descriptors, by relying on images for image convolutions, building on the strengths of the leading existing detectors and descriptors, and simplifying the relevant methods. Juan and Gwun (2009) presented a detailed comparison for SIFT, PCA-SIFT, and SURF. The comparison was based on David's algorithm. van de Sande *et al.* (2010) used color SIFT to increase the illumination invariance and discriminative power. Other researchers (Liu *et al.*, 2008; Vedaldi and Fulkerson, 2010) have used SIFT descriptors at every image location and obtained promising results in many applications; this method is called dense SIFT. Liu *et al.* (2008) aligned two images by sampling their common dense SIFT features. Their approach can robustly align complex scenes with large spatial distortions. Vedaldi and Fulkerson (2010) demonstrated that dense SIFT could more rapidly produce descriptors that are equivalent to those produced by SIFT. We will improve the performance of dense SIFT by using it in salient regions to reduce background interference.

Methods for detecting salient objects in images have received considerable attention in recent years because of their broad range of applications. The goal of salient-object detection is to find the most informative and important regions in images. The methods can be applied to image segmentation (Goferman *et al.*, 2010), image cropping (Santella *et al.*, 2006), object detection and recognition (Kanan and Cottrell, 2010; Rutishauser *et al.*, 2004), picture collage (Goferman *et al.*, 2010), etc. Salient-object detection typically involves computing the saliency map and outputting the salient object. Early studies on salient-object detection obtained the saliency map using visual attention. Methods include the bottom-up model of visual attention (Itti and Koch, 2001) and the top-down approach (Kanan *et al.*, 2009). Goferman *et al.* (2012) proposed 'context-aware saliency' to detect image regions that represent a scene. This technique is based on four principles observed in psychological literature, and produces promising results. However, most techniques are computationally expensive, and thus they are time-consuming and cannot be applied in most industrial fields. Rahtu *et al.* (2010) combined a saliency measure with a conditional-random-field (CRF) model and obtained promising results. Based on their work, we propose an adaptive optimization algorithm for determining the feature map, which better retains the contours of salient objects and reduces the computational time.

Felzenszwalb *et al.* (2010b)'s deformable parts model (DPM) is one of the most popular object detection methods. It is the foundation of the winning system in Pascal VOC 2007-2011. However, it is often important to use large training sets for objects with many rigid components and highly accurate results are difficult to achieve. Our work alleviates these difficulties, as discussed in Section 3. Based on the pictorial structure framework (Fischler and Elschlager, 1973; Felzenszwalb and Huttenlocher, 2005), our method increases the outlines of patches from the coarse detection phase.

Our formulation consists of two phases: coarse detection and refinement classification. The coarse

detection phase frees us from high-resolution images and focuses the technique on small image patches of objects. The refinement classification accurately determines which patches are objects.

2 Methods

Fig. 1 shows the architecture of the proposed formulation. First, we perform a coarse detection in high-resolution images to find objects with HOG features that are similar to our objects. Because the detection typically yields several small patches and does not include the entire object, we expand the outline based on part models. Second, we extract these image patches and generate their dense SIFT features to determine whether they are objects. A multi-detector is used for the coarse detection stage, and a multi-classifier is used in the refinement phase. Finally, we mark the objects in the original images and categorize the images according to the presence of objects.

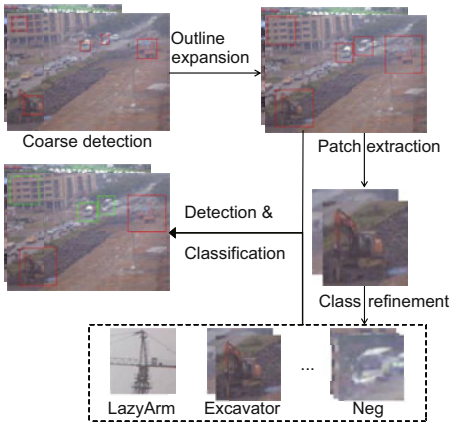

Fig. 1 An overview of the proposed method

2.1 Coarse detection with color frequencies

We considered the CIELab, RGB, and HSV color spaces. The HOG descriptors are concatenated with all channels of the color space. CIELab performed the best, HSV was almost equal to RGB, and they all performed better than the gray-HOG descriptor. However, the length of the features and the computational complexity increased linearly with the number of color channels.

Most same-category objects can be clustered according to color (particularly for engineering vehicles), which is an advantage for detection algorithms. Considering this, we call an object's color information its 'color frequency features'. We have concatenated the color frequency features with the HOG descriptors to improve the detection accuracy. The purpose of coarse detection is to achieve a high detection rate in a small amount of time. A high false positive rate is acceptable. There are negligible differences between the false positive rates using CIELab and our color frequency method when we are considering high false positive rates (Fig. 2). However, our method is very quick. Fig. 3 illustrates the concatenation of these features into a classical HOG descriptor.

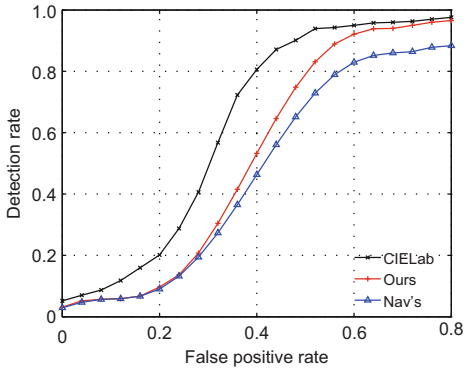

Fig. 2 Comparison of our HOG descriptor, the CIELab descriptor, and the classical descriptor

We compute the horizontal ($\boldsymbol{G}_{\mathrm{H}_i}$) and vertical ($\boldsymbol{G}_{\mathrm{V}_i}$) gradients of these images in three dimensions by applying the filter [−1 0 1] (Fig. 3b). Here, i denotes the ith-dimensional color space. We then find the maximum gradient of the three dimensions at the same pixel using

$$\boldsymbol{M}(x,y)=\max_i \sqrt{\boldsymbol{G}_{\mathrm{H}_i}(x,y)^2+\boldsymbol{G}_{\mathrm{V}_i}(x,y)^2}, \quad i=1,2,3. \tag{1}$$

These three features improve the accuracy when objects are clustered according to color. There will be redundancy when objects are not clustered by their color, but there is no adverse impact on the accuracy. We also record the dimension that contains the maximum value and compute the orientation of the gradient using

$$\theta(x,y)=\arctan\frac{\boldsymbol{G}_{\mathrm{H}}(x,y)}{\boldsymbol{G}_{\mathrm{V}}(x,y)}. \tag{2}$$

The norm and orientation of the gradient of the image are computed using the above method.

We create the selected color dimension in this step. Then the image is split into cells (C), as shown in Fig. 3b. Each cell is split into N bins and we compute a histogram (Fig. 3c). W is a detected window, the histograms of which determine a final descriptor. There is a block B in W, where B is composed of several cells. All histograms within a block are normalized and concatenated with the color frequency (Fig. 3d). The sliding block B in window W with stride S creates the histograms of W. A final descriptor is obtained by grouping all the normalized histograms of W into a single vector.

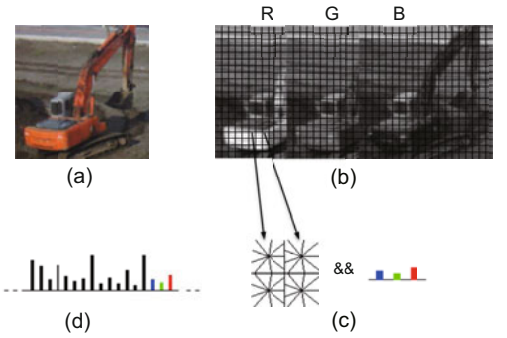

Fig. 3 HOG descriptor with color frequencies: (a) an image patch; (b) the RGB color spaces of (a); (c) the HOG features and color frequencies of (b); (d) the HOG descriptor added with color frequencies. References to color refer to the online version of this figure

Fig. 4 compares our algorithm with that of Dalal and Triggs (2005) in terms of descriptor length. Our descriptor is longer than that of Dalal and Triggs (2005), when the image is relatively large. However, there is a negligible time difference if the descriptor length is constrained to 6000. If we consider the linear SVM as a baseline classifier, our improved descriptor has a higher detection rate (Fig. 2). Because in the subsequent steps we apply a refinement classification, our improvement of the detection rate, which results in an error rate of greater than 30%, can help improve the whole system precision.

Our HOG descriptors focus on the rigid components because there may be frequent changes to the relative positions of components. Many image regions are obtained using these descriptors and a linear SVM detector, which may or may not contain object components. We then expand the regions so that they contain entire objects to produce a precise classification.

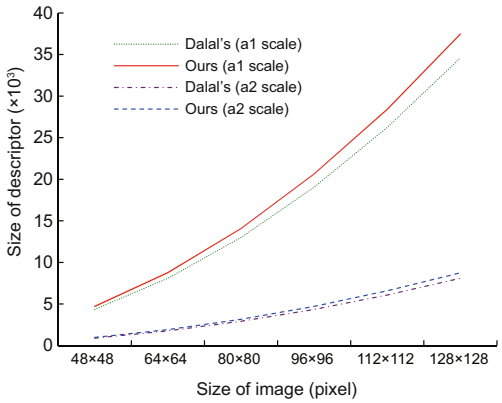

Fig. 4 Comparison of descriptor lengths. For the a1 scale, the sizes of cell C, block B, and stride S are 4×4, 8×8, and 4×4, respectively. For the a2 scale, the sizes of cell C, block B, and stride S are 8×8, 16×16, and 8×8, respectively

2.2 Outline expansion based on part models

Felzenszwalb *et al.* (2010b) used a star-structured part-based model, which was defined with a 'root' filter and a set of part filters and deformation models. Here, we define the rigid components of the object as the goals of coarse detection and 'root' filter models (Fig. 5).

Fig. 5 Expanding a patch's contour (the cab is the goal part and the array R is [2.0, 2.0, 2.0, 1.0])

We train only the root filters, and detect the main components using the method described in Section 2.1 (coarse detection). We increase the outline of each goal part according to the possible relative positions of the rigid components, using the goal part as the point of reference. The outline is expanded using the following steps:

1. Construct an array P that stores the upper-left and lower-right coordinates of the goal, which is obtained using the method given in Section 2.1.

2. Estimate four parameters that represent the ratios of four possible components (left, top, right, bottom) of the goal, and store them (left, top, right, bottom) in array R successively.

3. Increase the outline according to the size of

the goal and the four directional ratios, and store the new coordinates in array E.

This process is summarized as

$$E[i] = \alpha \cdot |P[(i+2)\%4] - P[i]| \cdot R[i] + P[i], \quad i = 0, 1, 2, 3, \tag{3}$$

where % is the 'mod' operator, and R can be easily estimated when collecting the training set. E should be adjusted to ensure that all the positions are contained in the images (Fig. 5). Errors in the patch contours are acceptable, because we reduce them in the subsequent step.

2.3 Saliency with adaptive acceleration

The function of this part of our technique is to extract salient objects from image patches, to reduce the computational cost of our SIFT extraction method and improve the accuracy of the classifier.

In Rahtu *et al.* (2010), the saliency measure applies a sliding window to the image and compares the contrast in each window according to the distribution of certain features in an inner window and the distribution in the edge of the window. They considered a rectangular window (W) that was divided into two disjoint parts: a rectangular inner window K and a border B. They hypothesized that the points in K were salient and the points in B were part of the background. Let Z be a random variable with values in W, which describes the distribution of pixels in W. Then the saliency measure of a point is defined as the conditional probability

$$S_0(x) = P(Z \in K | F(Z) \in Q_{F(x)}), \tag{4}$$

where x denotes every point in the image, F is a map that maps x to a certain feature $F(x)$, and the feature space is divided into disjoint bins. $Q_{F(x)}$ denotes the bin that contains the feature $F(x)$. The saliency measure of x is always a number between 0 and 1. If the feature at x is similar to the features at points in the inner window, pixel x is salient. We can compute $S_0(x)$ using Bayes' formula, that is,

$$S_0(x) = \frac{P(F(x)|I_0)P(I_0)}{P(F(x)|I_0)P(I_0) + P(F(x)|I_1)P(I_1)}, \tag{5}$$

where I_0 represents the condition $Z \in K$, I_1 represents $Z \in B$, and $F(x)$ denotes $F(Z) \in Q_{F(x)}$.

Using the CIELab color model, a regularized histogram for $P(F(Z) \in Q_{F(x)} | Z \in K)$ is defined based on a Gaussian function, to increase the robustness of F. The saliency map is achieved by sliding window W (with different scales) over the image, and the final saliency value is the maximum over all windows that contain the pixel. This implies that an increase in the scale results in an increase in accuracy for some methods. Four scales were applied in the experiments in Rahtu *et al.* (2010).

We reduced the number of scales to two to extract our salient object by optimizing the saliency value. Consider a matrix $\boldsymbol{M}$ that stores a saliency map in one scale, which has values between 0 and 1. We reassign the values to either t_1 or t_2, where t_1 is the value of the background and t_2 the value of the salient object, in a similar way to that in Otsu (1975). Uniformly divide the range [0, 1) into 100 parts (levels), i.e., [0, 0.01, ..., 0.99]. Denote N the total number of points in the matrix and n_i the number of points at level i. The probability of level i is

$$p_i = \frac{n_i}{N}, \quad \sum_{i=0}^{1} p_i = 1. \tag{6}$$

The matrix points are split into two classes using a threshold of t: with levels [0, 0.01, 0.02, ..., t] and with levels [t+0.01, t+0.02, ..., 0.99]. The between-class standard deviation for the two classes is

$$\sigma(t) = \left[P_1 \left(\frac{1}{P_1} \sum_{i=0}^{t} i p_i - \sum_{i=0}^{1} i p_i \right)^2 + P_2 \left(\frac{1}{P_2} \sum_{i=t+0.01}^{0.99} i p_i - \sum_{i=0}^{1} i p_i \right)^2 \right]^{1/2}, \tag{7}$$

where $P_1 = \sum_{i=0}^{t} p_i$ and $P_2 = \sum_{i=t+0.01}^{0.99} p_i$. We compute the between-class standard deviations for all levels between 0 and 1. We select the optimal threshold (t^*) that maximizes the between-class standard deviation. For these two optimal classes, we re-apply the processing and find the optimal thresholds (t_1 and t_2) of the two classes, respectively. The values of $\boldsymbol{M}$ are reassigned to t_1 when they are less than t^* and to t_2 when they are greater. After this operation, there are only two values in the saliency map, t_1 and t_2. Using two scales, we obtain two saliency maps. The final saliency value is the maximum of these two saliency maps.

Compared with Rahtu *et al.* (2010), our saliency map's foreground is sharper and the outline of the salient object is more complete (Fig. 6). For these

comparisons the Caltech 101 (Li *et al.*, 2007) and engineering vehicles datasets were used. For Caltech 101, the average ratio of Rahtu *et al.* (2010)'s salient areas to ours is 0.90, and it is 0.87 for the engineering vehicle dataset. We ran these experiments more than 10 times, and randomly chose 2000 samples from the dataset each time. Fig. 7 also shows that we detected more of the bodies of the engineering vehicles.

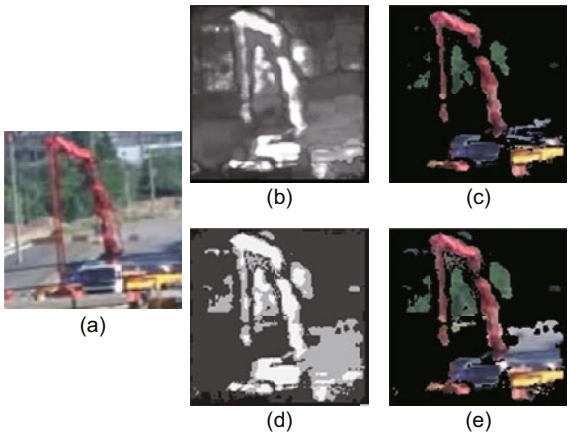

Fig. 6 Comparison of the original and accelerated saliency maps: (a) original image; (b) Rahtu *et al.* (2010)'s saliency; (c) object detected by Rahtu *et al.* (2010); (d) our saliency; (e) object detected by our saliency

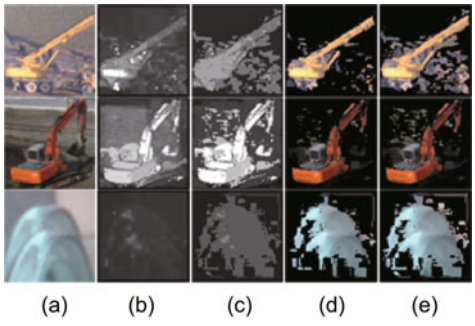

Fig. 7 Several comparison examples: (a) original images; (b) Rahtu *et al.* (2010)'s saliency images; (c) our saliency images; (d) the objects detected by Rahtu *et al.* (2010)'s method; (e) the objects detected by our method

Our method better satisfies our requirements; it pays more attention to the integrity of the object and can tolerate a certain number of saliency errors. Moreover, it is twice as fast as the method of Rahtu *et al.* (2010).

In our experiments the ratios of the row and column sizes and the sampling steps in the two scales to the largest image dimension are {0.2, 0.2, 0.02; 0.5, 0.5, 0.03}, respectively. We set the ratios of the row and column of the border B to the scale window W to {0.1, 0.1} in both scales. The ratios can be found by testing all possible window positions and scales with an appropriate growth span. Here, the growth span is 0.01 when it comes to the sampling steps. Otherwise, it is 0.1.

2.4 Dense SIFT with a saliency mask

Dense SIFT extracts SIFT features from a regular dense grid of the image. The main concept is to divide the image into density collections of independent patches, sample each patch for a SIFT vector, and combine the vectors into the image's descriptor. By enriching the feature vectors, we can improve the description of the image information used in Dalal and Triggs (2005)'s method.

In a similar way to Liu *et al.* (2008), we divide an image into many small patches with equal scales. We define the center pixel of each patch as the key point, and compute a SIFT feature for each patch. The patch is divided into 16 components that contain 4×4 image grids. We calculate an orientation histogram of the gradient with eight bins, which covers the 360-degree range of rotations for each component. When distributing each gradient value into neighboring bins, we use a trilinear interpolation to avoid the boundary effect of histogram binning. The SIFT feature of the patch is obtained by combining the 4×4×8 bins. Thus, the dimensionality of a SIFT feature is 128. The SIFT descriptor of the image is generated by combining the SIFT features of all the patches.

This classical dense SIFT computes all the patches, without considering unnecessary computations and interferences from complex backgrounds. After analyzing the performance of the saliency mask from Section 2.3, we apply it before computing the SIFT features of the patches. This significantly reduces the background interferences and saves time.

Fig. 8 shows two strategies for applying the saliency mask to extract dense SIFT features, which is called the saliency mask method. The image is divided into cells similar to the HOG method. A block containing cells is defined as a patch. Here, the stride of the block is equal to the size of cells. All patches are obtained when the block scans over the image. In Figs. 8a, 8a1, 8b1, and 8c1, the image contains 120×120 pixels, the cell contains 8×8

pixels, and the block contains 8×8 pixels. There are 255 patches with 95 effective patches being generated. In Figs. 8a, 8a2, 8b2, and 8c2, the cell contains 4×4 pixels and the block contains 8×8 pixels. There are 841 patches with 402 foreground patches being obtained.

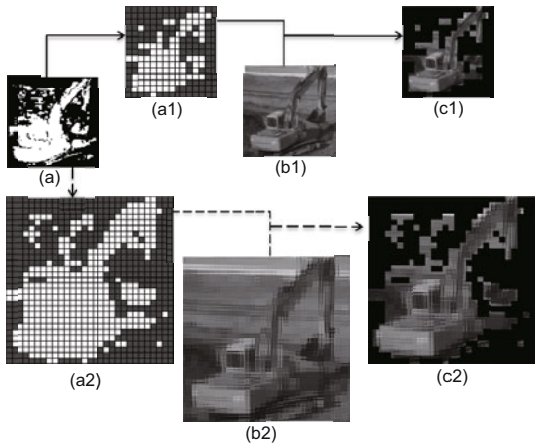

Fig. 8 Process of using the saliency mask. There are two strategies: (a) is the binary saliency map, (a1) and (a2) are path maps of the binary saliency map, (b1) and (b2) are patch maps of the gray image, and (c1) and (c2) are saliency patch images

First, the binary saliency map of the image is generated based on the method described in Section 2.3 (Fig. 8a). Second, P is created as the patch map of the binary saliency map (Figs. 8a1 and 8a2). All patches of the binary saliency map are generated using the saliency mask method. If the center pixel of the patch is black and the sum of the black pixels in the patch is greater than a threshold (γ), this patch is defined as the background and all pixels of the patch are assigned 0. Otherwise, it is the foreground and all pixels are assigned 255. In these experiments, we set the threshold γ to 0.6 times the number of pixels in the patch. After this processing, all patches are combined to generate P. Third, the patch map of the gray image (G) is created in a similar way (Figs. 8b1 and 8b2). Then the saliency patch image is created based on P and G (Figs. 8c1 and 8c2). The patch map of the gray image (G) contains all patches of the gray image and the patch map of the binary saliency map (P) determines whether the patch is in the foreground or background. Finally, the dense SIFT descriptor of this image is obtained by combining all the SIFT features of the foreground patches in the saliency patch image.

2.5 Bag-of-words (BOW) model with spatial pyramid and histogram intersection kernel (HIK)

A spatial pyramid (Lazebnik *et al.*, 2006) and histogram intersection kernel (HIK) (Grauman and Darrell, 2005) can be applied to improve the performance when considering a bag-of-words model (BOW) (Li and Perona, 2005). Fig. 9 presents a comparison of four kernels, their optimizations using the spatial pyramid, and the saliency forms of two superior kernels. Fig. 9 was produced by randomly selecting 30 categories from Caltech 101 (Li *et al.*, 2007), and averaging the results of several experiments. We applied 5-fold cross-validation (Breiman and Spector, 1992) and the Wilcoxon rank-sum test (Wilcoxon, 1945) for statistical significance. There were 50 or 30 categories in the training data and 20 in the test dataset. Fig. 9 shows that the pyramid HIK with saliency was the most accurate, and thus we used this method in the remainder of this study.

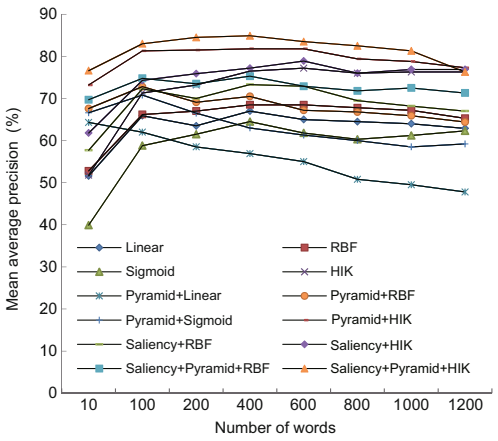

Fig. 9 Comparison of common kernels and their optimizations

Lazebnik *et al.* (2006) quantized the feature vectors into M discrete types, and subdivided the image at a resolution of L levels. The sum of the separate channel kernels is the final kernel, that is,

$$\boldsymbol{K}^L(X,Y)=\sum_{m=1}^{M}\boldsymbol{k}^L(X_m,Y_m), \tag{8}$$

where X_m and Y_m represent the coordinates of the features of type m that are found in the respective images. In practice, we set $L=2$. M is defined based on the size of the SIFT descriptor. Fig. 9 also shows the relationship between M and the SIFT descriptor

of Caltech 101, where most of the image patches have descriptor lengths in the range of 600–1200.

Considering Wu and Rehg (2009) and hypothesizing that the size of the training images' histograms' BOW was $N\times M$, we generated the images' intersection kernels by calculating the sum of the minimum of each row in the BOW histogram and the image's BOW. That is,

$$\boldsymbol{k}_{\mathrm{HI}}(\boldsymbol{h}_k)=\sum_{i=1}^{M}\min(\boldsymbol{h}_k,\boldsymbol{h}),\ k=1,2,\ldots,N, \quad (9)$$

where M is the channel of the original kernel, $\boldsymbol{h}$ is the histogram of the training images' BOW, and $\boldsymbol{h}_k$ is the histogram that contains the kth image's BOW in each row. By combining all the $\boldsymbol{k}_{\mathrm{HI}}$, we create the histogram intersection kernel for the BOW model. When testing, $\boldsymbol{h}_k$ contains the test image's BOW and $\boldsymbol{k}_{\mathrm{HI}}$ is the test image's kernel.

3 Experimental results

Fig. 9 validates our saliency method for Caltech 101 based on 5-fold cross-validation. The kernel with the best performance was HIK with a pyramid, which had a mean average precision (MAP) of 81.8% at 400 words. The saliency method increased the MAP to 84.9%. Because we obtained small patches with similar characteristics to Caltech 101 after the coarse detection process, we used Caltech 101 to build a saliency test even though it contains images with single-centered objects and smooth backgrounds.

We considered three datasets. One dataset contains monitoring data from power facilities, and contains 8000 images for each day. The goal of our work is to detect engineering vehicles in these images, such as 'cement car' (car of watering cement), crane, excavator, and 'lazyArm' (lazy arm). The second and third datasets were VOC 2012 and VOC 2011. Each dataset contains objects from 20 categories, which poses a challenge in object detection because there are significant variations in the appearances of objects from each category. We applied an SVM as our detector and classifier in all cases.

3.1 Remote monitoring dataset

3.1.1 Comparison with existing methods

Our work was aimed to detect engineering vehicles from monitoring images. In this dataset, 97% of the images were 1280×1024 pixels or larger, but the target objects covered only a few pixels. A refined detection made directly from these images would require more than 20 s of computational time for each image, which is unacceptably long. Thus, our method first uses a coarse detection method to extract image patches that may include object components. The main purpose of this coarse detection stage was to achieve a high detection rate in a short period of time. Because the vehicles' rigid components can change their relative positions, we detected only the main components in this phase (e.g., cabs and arms), as shown in Fig. 10.

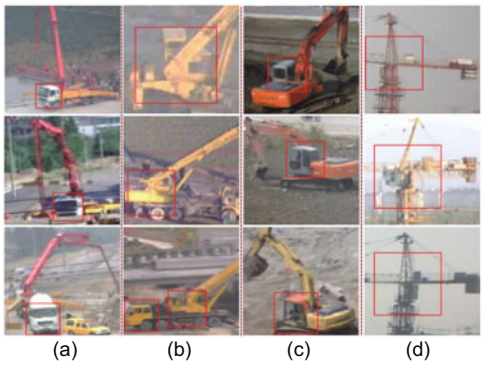

Fig. 10 Engineering vehicles with highlighted main parts: (a) cement cars; (b) cranes; (c) excavators; (d) lazy arms

Fig. 2 illustrates that the proposed method generates many false positive image patches to guarantee that the detection rate exceeds 0.9. We performed several experiments to determine the HOG descriptor parameters, and found that the best detection window (W) was 40×40 pixels with 18 bins.

After the coarse detection stage, we obtained many small patches that may contain components of objects. We increased the size of these patches so that they covered the entire objects using the method described in Section 2.2. We generated a saliency mask to reduce interference and extract the SIFT features. Then we applied an SVM multi-classifier to distinguish chaotic patches. Fig. 11 validates our saliency method by comparing it with the radial basis function (RBF) and HIK. The proposed saliency method improved the accuracy, and reduced the operational load by decreasing the descriptor length.

Table 1 contains the results of the refinement classification phase. The average accuracy was 91.0%. Table 2 presents the final detection

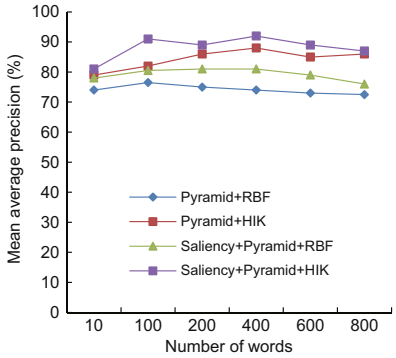

Fig. 11 Comparison of kernels in the dataset

results, and compares our method to a classical HOG without the saliency model and DPM (Felzenszwalb *et al.*, 2010b). We randomly sampled the tests and applied 10-fold cross-validation (Breiman and Spector, 1992). The stated results are the average of all tests. In this application, the false positive rate was 0.6 to ensure that the detection rate in the coarse phase was approximately 0.96 (Fig. 2).

Table 1 Accuracy of our classification in the second phase (refinement classification phase) for the engineering vehicle dataset (%)

	CC	Crane	ET	LazyArm	Neg
CC	87.1	0.0	10.3	2.6	0.0
Crane	14.2	82.6	3.3	0.0	0.0
ET	4.9	4.3	88.1	0.0	2.7
LazyArm	0.0	0.0	1.7	97.0	1.3
Neg	0.0	0.0	0.0	0.0	100.0

CC: cement car; ET: excavator

Table 2 Comparison of the proposed method with existing methods for the engineering vehicle dataset (%)

Category	Ours 0.96 rate	Ours 0.92 rate	HOG	DPM 0.96 rate
Cement car	83.7	80.2	75.7	77.5
Crane	79.5	76.2	72.3	75.9
Excavator	84.6	81.1	76.2	81.8
LazyArm	93.1	89.2	84.6	92.7

'Ours 0.96 rate' means that the detection rate in the first phase is set to 0.96; 'Ours 0.92 rate' sets the detection rate to 0.92; 'HOG' applies the classical gray-HOG descriptor in the first phase and the classical SIFT descriptor only with the pyramid HIK in the second phase; 'DPM 0.96 rate' uses the DPM method (Felzenszwalb *et al.*, 2010b) with a detection rate of 0.96

3.1.2 An emphasis on comparison with DPM

When the relative positions of object components can significantly vary (e.g., in engineering vehicles), the accuracy of the DPM method strongly depends on having a sufficiently large training set for the number of sub-models. However, it is hard to determine the size of the required training set, and it is impossible to collect large training sets for real applications. Maximizing over latent part locations in latent SVMs is an essential part of the DPM. However, significant variations in the relative positions of the components reduce its reliability. Our work can avoid these weaknesses with two thousand to three thousand positive samples in each category. We first detected the main components of objects, and then determined bounding boxes based on the models. Finally, we applied the bag-of-features (Zhang *et al.*, 2007) idea to obtain a precise classifier.

We also performed experiments to test DPM performance on our dataset. Different sample sizes and 10-fold cross-validation were used. The stated results were the average of all the experiments. Fig. 12 shows the results for the excavator dataset, and confirms that DPM strongly depends on the number of training sets and models. DPM performs well using the largest training set. However, our method outperforms DPM (Table 2).

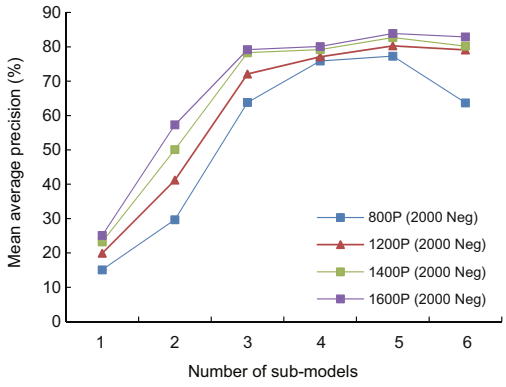

Fig. 12 Comparison of DPM's results with training sets of different sizes, with the detection rate being set to 0.96 (Table 1). '1600P (2000 Neg)' implies that the training set has 1600 positive samples and 2000 negative samples

Felzenszwalb *et al.* (2010b)'s method requires more than 15 s per image in the dataset (without parallelization), with five components per image. Researchers have recently accelerated DPM by one order of magnitude. Modified methods include cascade (Felzenszwalb *et al.*, 2010a), coarse-to-fine (Pedersoli *et al.*, 2015), branch-and-bound (Kokkinos, 2011), and fast Fourier transform (FFT)

(Dubout and Fleuret, 2012). Yan *et al.* (2014)'s method takes 3 or 4 frames/s for a given category with six components per image, on the Pascal VOC. This is currently the fastest method available. Our work ran at 1 frame/s per image on Pascal VOC, and at 0.5 frames/s per image for the engineering vehicle dataset. This is faster than most DPM algorithms. Our PC has a 2.5 GHz Intel Core i5 CPU, and we used only one thread.

3.2 Generalization performance

In this part, we chose the VOC 2012 and VOC 2011 to test our formulation's generalization performance. Our methods can perform better in some VOC 2012 and VOC 2011 categories compared to the challenge winners. Following is the detailed experiment on the PASCAL VOC 2012.

The 20 categories in the dataset contain significant variations. The resolutions of most images are from 500×300 or 300×500 to 500×500 pixels. The ratio of the area of positive objects also varies significantly from 0.12 to 1.00. We resized the original images to 50% and 200%, respectively. The samples contained all the original images. In the coarse detection stage, we used the images from one folder of each category to train the HOG descriptor, and used the images from another folder as the test set.

When increasing the outline, the array (R) of the four parameters should be easily estimated during the sample collection phase. The parameters were not very precise, but this is acceptable because the saliency work in the subsequent steps can reduce the errors. R was [0.5, 1.0, 0.5, 1.0] for the airplane, bicycle, boat, bus, car, motorbike, and train categories, [3.0, 1.0, 3.0, 3.0] for the bird, cat, cow, dog, horse, and sheep categories, [0.5, 0.5, 0.5, 0.5] for the bottle and person categories, and [2.0, 1.0, 2.0, 1.0] for the chair, dining table, potted plant, soft, and TV monitor categories.

In the refinement phase, we calculated a binary classification for the 20 categories. For each category, the negative samples were patches of other samples and some randomly extracted background patches. The main objective of this paper was to detect images that contained the target object, and thus the object contours were not very accurate. Table 3 shows our results based on this evaluation criterion and a qualitative comparison with the VOC 2012 winner. Table 3 illustrates that our formulation performed well for objects with similar HOG feature parts, such as the airplane, car, and person categories. However, for objects that have parts that can significantly vary (such as the bird, boat, and chair categories), there was a decrease in performance. Our methods also performed better in some VOC 2011 categories such as airplane, bicycle, cow, motorbike, and sheep. For brevity, we do not provide the VOC 2011 details. However, these results can be provided upon request.

Table 3 Results of our method and a comparison with the VOC 2012 winner (%)

Category	Our method			VOC 2012 winner
	Coarse detection phase*	Refinement classification phase	Entire detection method	
Airplane	82.3	89.5	**73.7**	65.0
Bicycle	70.1	84.8	**59.4**	54.5
Bird	33.8	70.5	23.8	**25.1**
Boat	29.3	72.9	21.3	**24.9**
Bottle	46.5	72.7	**33.8**	32.1
Bus	69.2	82.5	57.1	**57.1**
Car	68.7	81.7	**56.1**	49.3
Cat	60.2	83.3	50.1	**53.7**
Chair	28.5	60.3	17.2	**19.5**
Cow	47.8	78.3	**37.4**	35.3
Dining table	32.3	67.7	21.9	**38.1**
Dog	54.3	76.8	41.7	**42.9**
Horse	62.5	78.3	48.9	**51.0**
Motorbike	62.2	83.5	51.9	**59.5**
Person	72.2	84.3	**60.9**	46.1
PP	38.1	52.1	19.9	**22.8**
Sheep	60.6	71.9	**43.6**	40.3
Sofa	34.9	62.3	21.7	**39.7**
Train	58.3	83.8	48.8	**51.1**
TM	65.5	72.3	47.2	**49.4**
MAP	54.4	75.7	42.5	**42.9**

PP: potted plant; TM: TV monitor; MAP: mean average precision. A bold value means it is better. Though the method of the results of the VOC 2012 winner is not the same as the method of our results, we compared them because the detection rate of ours can be set in the coarse detection phase. * The false positive rate was set to 0.6

4 Discussion and conclusions

In this paper, we proposed a new method for detecting objects. The method simulates human beings' behavior in looking for a target object from hundreds of different classes of objects. A person will browse these objects quickly. When a similar object appears, he/she will spend a little more time to confirm whether the object is the target. We divided the process into two steps, coarse detection and

accurate confirmation, and presented the formulation. It can perform well in varying environments, particularly in datasets whose objects have components with significantly varying relative positions.

Our technique has a coarse detection phase and a refinement classification phase. We increased the detection rate in the coarse phase by adding color frequencies to the HOG descriptor. Because the operational costs increase as the descriptors grow, the length should be selected to balance the costs and detection rate. The contour expansion guarantees the integrity of the positive objects. We applied the saliency method to determine accurate contours and reduce background interferences. After acceleration, the saliency method was fast and accurate. Dense SIFT with a saliency pyramid HIK kernel was the most accurate.

In our experiments, we selected three datasets to analyze the proposed method. The results were very good for the first dataset, and our technique outperformed existing methods when applied to the VOC 2012 and VOC 2011 datasets.

The bottleneck in the total performance of the proposed method is the coarse detection rate. The HOG descriptor is fast and accurate when the object contours do not significantly vary. However, few objects satisfy this condition. We are currently attempting to combine this method with a deep neural network. We are also planning to investigate some more efficient and robust descriptors based on local binary pattern (LBP) (Ojala *et al.*, 2002), binary robust independent elementary features (BRIEF) (Calonder *et al.*, 2010), binary robust invariant scalable keypoints (BRISK) (Leutenegger *et al.*, 2011), and Daisy (Tola *et al.*, 2010), among others.

References

Avidan, S., 2006. SpatialBoost: adding spatial reasoning to AdaBoost. Proc. 9th European Conf. on Computer Vision, p.386-396. [doi:10.1007/11744085_30]

Bay, H., Ess, A., Tuytelaars, T., *et al.*, 2008. Speeded-up robust features (SURF). *Comput. Vis. Image Understand.*, **110**(3):346-359. [doi:10.1016/j.cviu.2007.09.014]

Breiman, L., Spector, P., 1992. Submodel selection and evaluation in regression. The X-random case. *Int. Statist. Rev.*, **60**(3):291-319.

Calonder, M., Lepetit, V., Strecha, C., *et al.*, 2010. BRIEF: binary robust independent elementary features. Proc. 11th European Conf. on Computer Vision, p.778-792. [doi:10.1007/978-3-642-15561-1_56]

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.886-893. [doi:10.1109/CVPR.2005.177]

Déniz, O., Bueno, G., Salido, J., *et al.*, 2011. Face recognition using histograms of oriented gradients. *Patt. Recogn. Lett.*, **32**(12):1598-1603. [doi:10.1016/j.patrec.2011.01.004]

Dubout, C., Fleuret, F., 2012. Exact acceleration of linear object detectors. Proc. 12th European Conf. on Computer Vision, p.301-311. [doi:10.1007/978-3-642-33712-3_22]

Felzenszwalb, P.F., Huttenlocher, D.P., 2005. Pictorial structures for object recognition. *Int. J. Comput. Vis.*, **61**(1):55-79. [doi:10.1023/B:VISI.0000042934.15159.49]

Felzenszwalb, P.F., Girshick, R.B., McAllester, D., 2010a. Cascade object detection with deformable part models. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, p.2241-2248. [doi:10.1109/CVPR.2010.5539906]

Felzenszwalb, P.F., Girshick, R.B., McAllester, D., *et al.*, 2010b. Object detection with discriminatively trained part-based models. *IEEE Trans. Patt. Anal. Mach. Intell.*, **32**(9):1627-1645. [doi:10.1109/TPAMI.2009.167]

Fischler, M.A., Elschlager, R.A., 1973. The representation and matching of pictorial structures. *IEEE Trans. Comput.*, **22**(1):67-92.

Goferman, S., Tal, A., Zelnik-Manor, L., 2010. Puzzle-like collage. *Comput. Graph. For.*, **29**(2):459-468. [doi:10.1111/j.1467-8659.2009.01615.x]

Goferman, S., Zelnik-Manor, L., Tal, A., 2012. Context-aware saliency detection. *IEEE Trans. Patt. Anal. Mach. Intell.*, **34**(10):1915-1926. [doi:10.1109/TPAMI.2011.272]

Grauman, K., Darrell, T., 2005. The pyramid match kernel: discriminative classification with sets of image features. Proc. 10th IEEE Int. Conf. on Computer Vision, p.1458-1465. [doi:10.1109/ICCV.2005.239]

Itti, L., Koch, C., 2001. Computational modelling of visual attention. *Nat. Rev. Neurosci.*, **2**(3):194-203. [doi:10.1038/35058500]

Juan, L., Gwun, O., 2009. A comparison of SIFT, PCA-SIFT and SURF. *Int. J. Image Process.*, **3**(4):143-152.

Kanan, C., Cottrell, G., 2010. Robust classification of objects, faces, and flowers using natural image statistics. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, p.2472-2479. [doi:10.1109/CVPR.2010.5539947]

Kanan, C., Tong, M.H., Zhang, L., *et al.*, 2009. SUN: top-down saliency using natural statistics. *Vis. Cogn.*, **17**(6-7):979-1003. [doi:10.1080/13506280902771138]

Ke, Y., Sukthankar, R., 2004. PCA-SIFT: a more distinctive representation for local image descriptors. Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.506-513. [doi:10.1109/CVPR.2004.1315206]

Kobayashi, T., 2013. BFO meets HOG: feature extraction based on histograms of oriented p.d.f gradients for image classification. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, p.747-754. [doi:10.1109/CVPR.2013.102]

Kokkinos, I., 2011. Rapid deformable object detection using dual-tree branch-and-bound. Advances in Neural Information Processing Systems, p.2681-2689.

Lazebnik, S., Schmid, C., Ponce, J., 2006. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.2169-2178. [doi:10.1109/CVPR.2006.68]

Leutenegger, S., Chli, M., Siegwart, R.Y., 2011. BRISK: binary robust invariant scalable keypoints. Proc. IEEE Int. Conf. on Computer Vision, p.2548-2555. [doi:10.1109/ICCV.2011.6126542]

Li, F.F., Perona, P., 2005. A Bayesian hierarchical model for learning natural scene categories. Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.524-531. [doi:10.1109/CVPR.2005.16]

Li, F.F., Fergus, R., Perona, P., 2007. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. *Comput. Vis. Image Understand.*, **106**(1):59-70. [doi:10.1016/j.cviu.2005.09.012]

Li, W.H., Lin, Y.F., Fu, B., *et al.*, 2013. Cascade classifier using combination of histograms of oriented gradients for rapid pedestrian detection. *J. Softw.*, **8**(1):71-77. [doi:10.4304/jsw.8.1.71-77]

Liu, C., Yuen, J., Torralba, A., *et al.*, 2008. SIFT flow: dense correspondence across different scenes. Proc. 10th European Conf. on Computer Vision, p.28-42. [doi:10.1007/978-3-540-88690-7_3]

Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.*, **60**(2):91-110. [doi:10.1023/B:VISI.0000029664.99615.94]

Ojala, T., Pietikainen, M., Maenpaa, T., 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Patt. Anal. Mach. Intell.*, **24**(7):971-987. [doi:10.1109/TPAMI.2002.1017623]

Otsu, N., 1975. A threshold selection method from gray-level histograms. *Automatica*, **11**:23-27.

Ott, P., Everingham, M., 2009. Implicit color segmentation features for pedestrian and object detection. Proc. IEEE 12th Int. Conf. on Computer vision, p.723-730. [doi:10.1109/ICCV.2009.5459238]

Pedersoli, M., Vedaldi, A., Gonzàlez, J., *et al.*, 2015. A coarse-to-fine approach for fast deformable object detection. *Patt. Recogn.*, **48**(5):1844-1853. [doi:10.1016/j.patcog.2014.11.006]

Rahtu, E., Kannala, J., Salo, M., *et al.*, 2010. Segmenting salient objects from images and videos. Proc. 11th European Conf. on Computer Vision, p.366-379. [doi:10.1007/978-3-642-15555-0_27]

Rutishauser, U., Walther, D., Koch, C., *et al.*, 2004. Is bottom-up attention useful for object recognition? Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.37-44. [doi:10.1109/CVPR.2004.1315142]

Santella, A., Agrawala, M., DeCarlo, D., *et al.*, 2006. Gaze-based interaction for semi-automatic photo cropping. Proc. SIGCHI Conf. on Human Factors in Computing Systems, p.771-780. [doi:10.1145/1124772.1124886]

Tola, E., Lepetit, V., Fua, P., 2010. DAISY: an efficient dense descriptor applied to wide-baseline stereo. *IEEE Trans. Patt. Anal. Mach. Intell.*, **32**(5):815-830. [doi:10.1109/TPAMI.2009.77]

van de Sande, K.E.A., Gevers, T., Snoek, C.G.M., 2010. Evaluating color descriptors for object and scene recognition. *IEEE Trans. Patt. Anal. Mach. Intell.*, **32**(9):1582-1596. [doi:10.1109/TPAMI.2009.154]

Vedaldi, A., Fulkerson, B., 2010. VLFeat: an open and portable library of computer vision algorithms. Proc. Int. Conf. on Multimedia, p.1469-1472. [doi:10.1145/1873951.1874249]

Wilcoxon, F., 1945. Individual comparisons by ranking methods. *Biometr. Bull.*, **1**(6):80-83.

Wu, J.X., Rehg, J.M., 2009. Beyond the Euclidean distance: creating effective visual codebooks using the histogram intersection kernel. Proc. IEEE 12th Int. Conf. on Computer Vision, p.630-637. [doi:10.1109/ICCV.2009.5459178]

Yan, J.J., Lei, Z., Wen, L.Y., *et al.*, 2014. The fastest deformable part model for object detection. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, p.2497-2504. [doi:10.1109/CVPR.2014.320]

Zaklouta, F., Stanciulescu, B., 2014. Real-time traffic sign recognition in three stages. *Robot. Auton. Syst.*, **62**(1):16-24. [doi:10.1016/j.robot.2012.07.019]

Zhang, J., Marszałek, M., Lazebnik, S., *et al.*, 2007. Local features and kernels for classification of texture and object categories: a comprehensive study. *Int. J. Comput. Vis.*, **73**(2):213-238. [doi:10.1007/s11263-006-9794-4]