Patch-based Image Correlation with Rapid Filtering

Guodong Guo
Dept. of Math & Computer Science
North Carolina Central University

Charles R. Dyer
Computer Sciences Department
University of Wisconsin-Madison

Abstract

This paper describes a patch-based approach for rapid image correlation or template matching. By representing a template image with an ensemble of patches, the method is robust with respect to variations such as local appearance variation, partial occlusion, and scale changes. Rectangle filters are applied to each image patch for fast filtering based on the integral image representation. A new method is developed for feature dimension reduction by detecting the "salient" image structures given a single image. Experiments on a variety images show the success of the method in dealing with different variations in the test images. In terms of computation time, the approach is faster than traditional methods by up to two orders of magnitude and is at least three times faster than a fast implementation of normalized cross correlation.

1. Introduction

Many computer vision applications need to know whether a pre-defined template sub-image is contained within a test image. Template matching techniques involve the translation of the template to every possible position in the test image and the evaluation of a measure of the match between the template and the image at that position [13].

One common measure to compare the similarity of a template image, $t(i,j)$, and a test image, $f(i,j)$, is the sum of squared difference (SSD), defined by

$$SSD(m,n) = \sum_{i,j}[f(i,j) - t(i-m, j-n)]^2 \quad (1)$$

Another common measure is the normalized cross-correlation (NCC). The $NCC(m,n)$ is defined by

$$\frac{\sum_{i,j}[f(i,j) - \bar{f}_{mn}][t(i-m,j-n) - \bar{t}]}{\sqrt{\sum_{i,j}[f(i,j) - \bar{f}_{mn}]^2}\sqrt{\sum_{i,j}[t(i-m,j-n) - \bar{t}]^2}} \quad (2)$$

where $\bar{t}$ is the mean of the template image and $\bar{f}_{mn}$ is the mean of the test image $f(i,j)$ in the region centered at (m,n).

A major disadvantage of both SSD and NCC measures is their computational cost when the template is large [13]. Hence, how to speed up correlation computations is critical for many applications.

For NCC, Lewis [5] proposed to use the integral image representation, which was first developed by Crow [1] for texture mapping, to compute the denominator in Eq. (2). The numerator in Eq. (2) can be computed by the fast Fourier transform (FFT). Viola and Jones [14] used the integral image for fast face detection. Jurie and Dhome [4] used the fast template matching for tracking. Schweitzer et al. [8] extended the integral image to compute the algebraic moments and approximated the image with low degree polynomials for fast template matching. Hel-Or and Hel-Or [3] used Walsh-Hadamard kernels and the integral image for fast feature extraction and matching. Tao et al. [10] approximated the template image by a linear combination of simple binary box features which can be computed efficiently by using the integral image representation. Although their matching is fast, it is quite slow to learn the non-orthogonal subspace based on their optimized orthogonal matching pursuit method [10]. Tang and Tao [9] adopted a matching pursuit method to search the non-orthogonal subspace, which is faster than that in [10].

Rosenfeld and Vanderbrug [7] proposed a multi-resolution scheme for template matching. The result in a low resolution image is refined at higher resolutions. Uenohara and Kanade [12] matched a large set of templates with the test image, and they represented the template set in a PCA subspace. An FFT was used to speed up correlation.

Almost all previous template matching methods focus on speeding up the computation, but have not addressed the occlusion problem, which is the issue of detecting the template when it is partially occluded in the test image. Variation in scale is another issue for template matching in practice, but little previous work has addressed it. The method in [3] can only handle patterns scaled by powers of 2 using Walsh-Hadamard kernels in SSD matching. A method in [9] can deal with scale differences but cannot handle the occlusion problem. In this paper, we propose a new approach for image correlation that is robust to partial occlusion and

1-4244-1180-7/07/$25.00 ©2007 IEEE

arbitrary scale change, in addition to fast matching.

While the SIFT method [6] can extract a sparse set of scale-invariant features for matching, the template matching methods that we used here are pixel patches. This general template matching is important when the images contain no rich texture information for local interest points extraction. Another scheme is to use an image pyramid for detecting templates at multiple scales in the test image, but this involves much more computation in searching over all possible scales.

1.1. Pixels, Patches, and Global Template

For numerous computer vision applications, the image can be analyzed at the patch level rather than at the individual pixel level. Patches contain contextual information and have advantages in terms of computation and generalization. For example, patch-based methods produce better results and are much faster than pixel-based methods for texture synthesis [2].

On the other hand, some computer vision applications require a large template to represent a complete object, especially for object recognition. For instance, face templates may contain the whole face for use in face detection [14] or face recognition [11]. Although large templates are useful for representing the whole object, they are sensitive to local variations and partial occlusion. To use large templates, an image can be represented by *an ensemble of patches*.

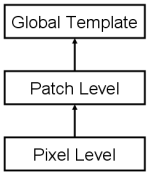


Figure 1. Three possible levels of methods for image correlation.

In image correlation or template matching, there are three levels of approaches: pixel level, patch level, and complete template, as illustrated in Figure 1. SSD and NCC measures use the whole template for matching but actually work at the pixel level (see Eqs. (1) and (2)). That is why standard SSD and NCC are slow and not robust to variations. Recently proposed methods [5] [10] [9] speed up NCC by using integral images for the denominator computation and deal with the numerator by a subspace approximation [10] [9] or FFT [5]. Methods in [7] [5] [12] [8] [3] [10] [9] are sensitive to local variations and occlusions because they used complete templates only.

1.2. Main Contributions

Motivated by the advantages of representing an image by a set of patches, we propose a patch-based method for image correlation. Our main contributions include: 1) proposing a new approach for template matching using an ensemble of patches instead of a single, large template; 2) developing a new method for feature selection given a single image; 3) addressing the issue of robustness in image correlation with respect to appearance variation, partial occlusion, and scale change all together.

The remainder of the paper is organized as follows. Section 2 introduces the patch-based matching criterion and rapid filtering for each patch. Section 3 describes a simple technique for feature selection given a single template image. Section 4 discusses the robustness of the method to different variations. Experimental results are presented in Section 5. Finally, conclusions are given in Section 6.

2. Patch-based Correlation

A template image is first decomposed into a set of patches. These patches can be overlapping or non-overlapping. In all our experiments, all patches are non-overlapping, as shown in Figure 2.

Figure 2. A template image (see Figure 5 (b)) is divided into non-overlapping patches.

The template image is now represented by $t = [p_1^t, p_2^t, \cdots, p_k^t]$ when it is divided into k patches. Then the similarity between template $t(i,j)$ and test image $f(i,j)$ is defined by

$$D(t, f, m, n) = \sum_{r=1}^{k} \| \, p_r^t - p_r^f(m,n) \, \| \qquad (3)$$

where $p_r^f(m,n)$ is the patch in the test image $f(i,j)$ corresponding to the patch p_r^t in the template, as the template $t(i,j)$ is translated to a position (m,n) in the test image. $\| \cdot \|$ is the norm, and we used the 1-norm in our experiments.

$D(t,f,m,n)$ is non-negative. The smaller the value of $D(t,f,m,n)$, the more similar the template t and the test image f at position (m,n). In our definition, the image correlation problem is to find the minimum of the objective function:

$$(m^*, n^*) = \arg\min_{m,n} D(t, f, m, n) \qquad (4)$$

Now the question is how to measure the similarity of patches, i.e., Eq. (3).

2.1. Rapid Filtering with Rectangle Filters

For each patch, we use a set of filters to extract features rather than directly using raw pixel intensities. There are two advantages to using features over raw pixels for image correlation: (1) a feature-based system operates much faster than a pixel-based system [14], and (2) features can encode some image structure information that can be used to reduce the number of features further, thus making the correlation process even faster (see Section 3).

A variety of filters can be used for filtering in each patch, including Gabor filters and Laplacian-of-Gaussian filters. Here we choose to use the rectangles filters which were originally used by Viola and Jones for rapid face detection [14]. One advantage of using rectangle filters is that they can be evaluated quickly with simple additions no matter how big the filters are, based on the integral image representation [1].

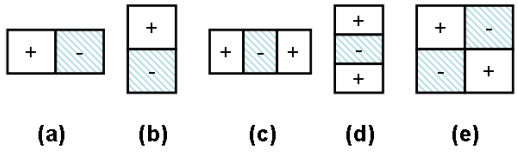


Figure 3. The filters applied to each image patch. The sum of the pixels that lie within the dark rectangles (with -) are subtracted from the sum of pixels in the light rectangles (with +). Filters (a) and (b) are useful for edge features, (c) and (d) for line features, and (e) for diagonal structures.

We employed five types of rectangle filters, as shown in Figure 3, similar to those used in [14]. The difference is that we apply the filters to each patch instead of the complete template image. These filters encode edge and line features in horizontal and vertical directions, or diagonal structures in images.

Provided with the filters, the similarity measure of Eq. (3), can be rewritten as follows:

$$D(t, f, m, n) = \sum_{r=1}^{k} \sum_{l=1}^{L} \parallel g_l \otimes p_r^t - g_l \otimes p_r^f(m, n) \parallel \quad (5)$$

where g_l are the rectangle filters, with $l \in \{1, 2, \cdots, L\}$, and $\otimes$ is convolution.

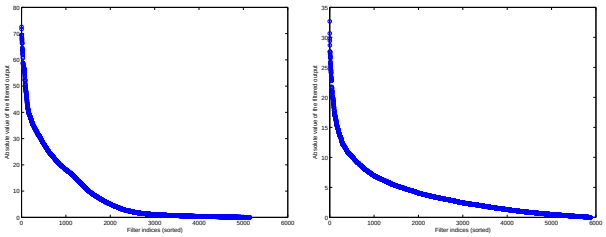
3. Image Structure based Feature Selection

As described in previous section, we apply rectangle filters to each patch. As a result, the number of features extracted from a template image is given by $\#features = \#patches \times \#filters$. Suppose the template is decomposed into 60 patches and 100 filters are applied to each patch, then the number of features will be 6,000. This number is usually smaller than the number of pixels in the template, e.g., 40,000 for a 200 x 200 image. Thus patch-based matching can be faster than pixel-based SSD or NCC. However, the number of extracted features is still too large and matching is not fast enough. So the next problem is how to reduce the number of features extracted from a template.

Feature selection is a classical problem in machine learning. Usually it requires a large number of training examples in each class. Classifiers are employed to evaluate the selected features on a validation set. For general template matching, however, it is not practical to collect many training examples. In addition, template matching based on SSD and NCC does not require any training examples.

In order to select features for further speeding up patch-based image correlation, we turn to the template image itself. Regions with "salient" image structures should play a more important role than "flat" regions for image correlation. But how to extract salient regions and relate them to feature selection? One intuitive way is to detect edges or corners, but this will involve more computations and it is not straightforward to relate the detected edges or corners to our features obtained by the rectangle filters.

Instead, we propose to detect "salient" image structures by directly comparing the filtered values. As introduced in the previous section, the rectangle filters we use respond strongly to image structures such as edges, lines, and diagonal structures. Here strong response means large values in the filtered results.



(a) (b)

Figure 4. (a) Filter responses on the template image of Figure 5 (b); (b) Filter responses on the template of Figure 6 (b).

This heuristic of detecting salient image structures is supported by real data. Figure 4 (a) and (b) illustrate the magnitude of the filtered values after sorting in descending order. They were extracted from two templates (Figure 5 (b) and 6 (b)), respectively. Based on observation in Figure 4 (a) and (b), we may choose a small number of features, e.g. the top 5%, for image correlation, and matching will be much faster.

Let $v_{l,r}^t = g_l \otimes p_r^t$ be the filtered value using the filter g_l on patch p_r^t. A small number of features can be selected by

the following measure:

$$v^t_{l',r'} > \alpha \times \max_{l,r} v^t_{l,r} \tag{6}$$

where α is a constant to control the number of selected features, and thus influences the computation time in matching.

4. Robustness

Our patch-based approach to image correlation has some special properties, especially its robustness with respect to different variations, in addition to its fast computation.

4.1. Appearance Variation

Since all patches are processed independently, any local appearance variation of objects in the test image only has influence on some patches, without causing a global effect on template matching. This can be seen from the definition of the similarity measure in Eq. (5). Furthermore, if the local variation is extremely large, one may assign smaller weights to those patches given some prior knowledge about the variation. For instance, for face objects, the variation in facial expression often produces large changes in the lower part of the face, so smaller weights can be used with patches located in the lower face.

The weighted similarity measure can be defined by

$$D(t, f, m, n, w) = \sum_{r=1}^{k} \sum_{l=1}^{L} w_r \parallel g_l \otimes p^t_r - g_l \otimes p^f_r(m, n) \parallel \tag{7}$$

4.2. Partial Occlusion

Partial occlusion is an issue in almost all computer vision applications. Our patch-based correlation method can effectively deal with partial occlusions. While the idea of weighting patches for local variations (see Eq. (7)) can also be adopted for partial occlusion, a simpler idea is to use the most similar patches in the test image for matching with the template, rather than assigning specific weights for each patch.

Let L^t, R^t, U^t, D^t denote the left, right, up, and down parts of the template, and $\Omega = \{L^t, R^t, U^t, D^t\}$. Assuming that the occlusion of an object in the test image is less than a half of the object, then we can always find at least one un-occluded part from Ω[1]. We measure the similarity of each part of the object and choose the most similar part to represent the whole similarity. The re-defined similarity measure is given by

$$D^o(t, f, m, n) = \min_{S \in \Omega} D(t, f, m, n, S) \tag{8}$$

[1]Here we ignore the occlusion in the middle of the object. But it is not difficult to extend the idea presented here to that case.

with

$$D(t, f, m, n, S) = \sum_{p^t_r \in S} \sum_{l=1}^{L} \parallel g_l \otimes p^t_r - g_l \otimes p^f_r(m, n) \parallel \tag{9}$$

4.3. Variable Scales

Objects may have different sizes in the template and test images. Traditional SSD and NCC matching does not consider scale changes. One may alter the test image into different sizes for matching with the template, but this will make the template matching process even slower. In contrast, the filtering in our approach can be evaluated in the test image at various scales without increasing the computation time significantly. The size of test images does not need to change. This computational advantage of rectangle filters based on the integral image representation was first demonstrated by Viola and Jones for face detection [14]. Our patch-based method can take the same approach, and thus can easily deal with variable scales in template matching.

5. Experiments

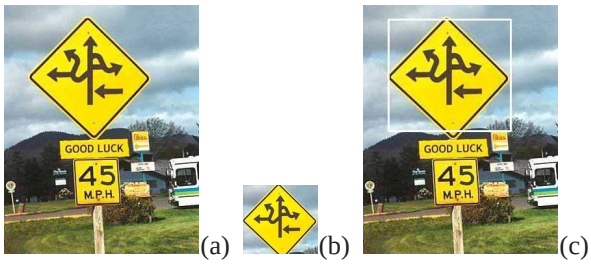
To evaluate our patch-based method for image correlation we conducted experiments on a variety of images. In all our experiments, the patch size is chosen as 24x24. Each template image is divided into non-overlapping patches of the same size, as shown in Figure 2. The patches close to the image boundary of the template were not be used if they were smaller than 24x24. The number of patches ranges from 10 to 56 in our experiments, depending on the size of the template image.

The filter size was chosen as 8x12 for (a) and (c), 12x8 for (b) and (d), and 8x8 for (e) in Figure 3. The filters are shifted by a step size of 4 pixels within each patch. As a result, there are 105 filters applied to each patch. For a template image with 56 patches, the resulting feature dimension is 5,880. To select the salient features, we chose $\alpha = 0.95$ in Eq.(6). The number of selected features is usually less than 30, therefore the matching is very fast. In all experiments, the template is translated with a step size of one pixel in the test image and only gray level intensity values are used for matching although color images are displayed.

Figure 5(a) is a road sign image. A sub-window containing the main sign is cropped from 5(a) and shown in 5(b), which is used as the template. Then our patch-based method is applied to match the two images. The correlation result is shown in Figure 5(c). The white box labels the position and size of the detected pattern in the test image which is correct when compared with the ground truth. The matching is fast and the computation time is given in Table 1.

Table 1. Comparison of the computation time (in seconds) of different methods on various images. Our patch-based method with feature selection is much faster than standard normalized cross correlation (NCC) and at least three times faster than a fast implementation of NCC using an FFT for the numerator and integral images for the denominator (F-NCC). The size of each test image and template is given under each experiment name. The symbols "o." and "s." are for occlusion and multi-scales, respectively.

	NCC	F-NCC	Ours
sign (300x384) (189x173)	24.39	3.03	0.95
fruit (512x480) (204x188)	109.53	13.55	3.77
face (640x486) (177x163)	137.16	16.91	4.95
sign(o.) (300x384) (189x173)	–	–	1.50
boat(s.) (710x505) (140x69)	–	–	9.67



(a) (b) (c)

Figure 5. (a) A road sign image (of size 300x384); (b) the template image (189x173); (c) matching result by our patch-based approach.

Figure 6(a) is a fruit image. A sub-window is cropped from 6(a) and shown in 6(b) which is used as the template. Figure 6(c) shows the detected pattern given by our patch-based method. The detected position is exactly the same as the ground truth.

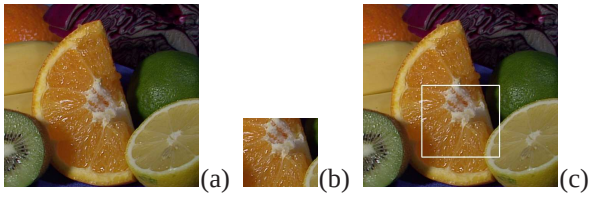

(a) (b) (c)

Figure 6. (a) A fruit image (512x480); (b) the template (204x188); (c) matching result by our method.

The experiments in Figures 5 and 6 demonstrate that the patch-based method works well for the given cropped templates. The next experiments will verify the robustness of the method with respect to different variations.

Appearance variation

Figure 7(a) is an image of a face with a neutral expression. Figure 7(b) is the face window cropped from 7(a). Applying the patch-based method to match 7(a) and 7(b), the face pattern is detected correctly and shown in Figure 7(c). More interestingly, the template is matched to another image of a smiling face of the same individual. The detected pattern of a smiling face is shown in Figure 7(d). There is no ground truth for measuring the positional accuracy of the smiling face, but one can visually check the matching result. It is quite accurate. This experiment shows the robustness of the method to local appearance change.

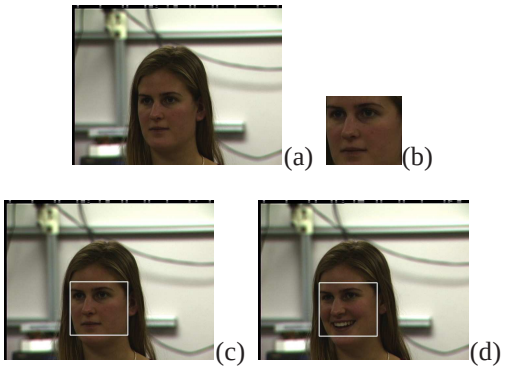

(a) (b) (c) (d)

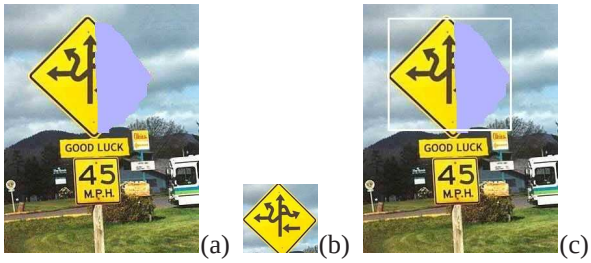
Figure 7. (a) A face image (640x486); (b) the template (177x163); (c) matching result; (d) the detected face pattern on a test image (640x486) of a smiling face.

Partial occlusion

To verify the robustness to occlusion, we manually erase the right half of the road sign in Figure 5(a) and show it in Figure 8(a). The matching template shown in Figure 8(b) is the same as Figure 5(b). For this example, we used the same weights for each patch. The partially occluded road sign is correctly identified by our patch-based method as shown in Figure 8(c). Previous methods using the global template cannot detect the occluded pattern as ours does. The reason is that they do not have a flexible mechanism to deal with partial occlusion.

Variable scales

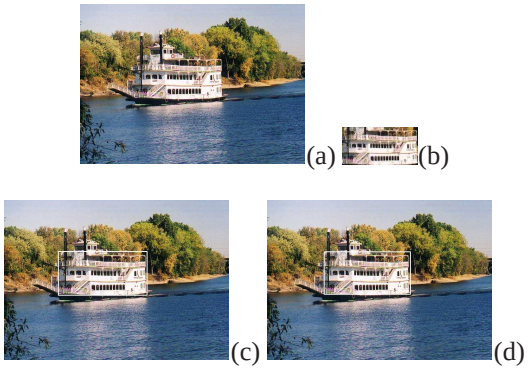
Another image variation is scale change. An object often appears with different sizes in different images. To verify the robustness of our method with respect to image scale change, a boat template is cropped from the image in Figure 9(a) and down-sampled by a factor of 2 in both directions. It is shown in Figure 9(b). Then we perform a multi-scale search of the template over the test image shown in Figure 9(a). Two schemes are executed in varying



(a) (b) (c)

Figure 8. (a) A partially occluded road sign image (300x384) used as a test; (b) the template image (189x173); (c) the detected pattern given by our method.

the scales: the first is to use scales of a power of 2, such as 1, 2, and 4. The correct position and scale found by the method are shown in Figure 9(c). The second scheme is to use a set of scales a factor of 1.25 apart. Then the detected scale is $1.25^3 = 1.95$. Figure 9(d) shows the detected pattern. Note that in the second scheme, the detected scale, position, and size of the boat pattern is slightly different from the ground truth, although the two results are almost the same visually. We intentionally chose different scales to search in order to verify the robustness of the method with respect to variable scales because in practice a method cannot exhaustively search over all continuous scales to guarantee the exact scale of the test image is not missed.



(a) (b) (c) (d)

Figure 9. (a) A boat image (710x505); (b) the template image cropped and down-sampled to (140x69); (c) the detected pattern using the power of 2 in scale search; (d) the identified pattern using a set of scales a factor of 1.25 apart.

Discussion

The patch-based image correlation method presented in this paper can be used for applications whenever standard SSD or NCC can be applied, e.g., object detection, recognition, and tracking, given only a single example image, i.e., the template, can be used. Because it is fast and robust with respect to many types of variations, a more interesting application of our method is to interactively collect data in a semi-supervised manner. The user provides a (cropped) template, and the matching technique finds similar patterns in a database and "cuts" them out for the user. This is easier than manually cropping patterns by the user for collecting training data for tasks in learning-based vision.

6. Conclusion

We have presented a new method for fast template matching. Our patch-based approach is robust to many types of variations, such as local appearance change, partial occlusion, and scale variation. To our knowledge, no previous methods address these variations all together in template matching. The rectangle filters applied to each patch can be evaluated quickly based on the integral image representation, and thus matching is faster than traditional approaches. A new feature selection strategy was developed based on detecting salient image structures that are encoded naturally by rectangle filters. Experiments on a variety of test images show that our patch-based correlation method is promising for fast template matching.

References

[1] F. Crow. Summed-area tables for texture mapping. *Computer Graphics*, 18(3):207–212, 1984.

[2] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In *SIGGRAPH*, pages 341–346, 2001.

[3] Y. Hel-Or and H. Hel-Or. Real-time pattern matching using projection kernels. *IEEE PAMI*, 27(9):1430–1445, 2005.

[4] F. Jurie and M. Dhome. Real time robust template matching. In *Proc. BMVC*, pages 123–132, 2002.

[5] J. P. Lewis. Fast template matching. *Vision Interface*, pages 120–123, 1995.

[6] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Computer Vision*, 60(2):91–110, 2004.

[7] A. Rosenfeld and G. Vanderbrug. Coarse-fine template matching. *IEEE Trans. on SMC*, 7:104–107, 1977.

[8] H. Schweitzer, J. Bell, and F. Wu. Very fast template matching. In *ECCV*, pages 358–372, 2002.

[9] F. Tang and H. Tao. Fast multi-scale template matching using binary features. In *IEEE WACV*, 2007.

[10] H. Tao, R. Crabb, and F. Tang. Non-orthogonal binary subspace and its applications in computer vision. In *Int. Conf. on Computer Vision*, pages 864–870, 2005.

[11] M. Turk and A. Pentland. Face recognition using eigenfaces. In *Proc. CVPR*, pages 586–591, 1991.

[12] M. Uenohara and T. Kanade. Use of fourier and karhunen-loeve decomposition for fast pattern matching with a large set of templates. *IEEE PAMI*, 19(8):891–898, 1997.

[13] D. Vernon. *Machine Vision: Automated Visual Inspection and Robot Vision*. Prentice Hall, 1991.

[14] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, pages 511–518, 2001.