Computation of Rotation Local Invariant Features using the Integral Image for Real Time Object Detection

Michael Villamizar[1], Alberto Sanfeliu[1] and Juan Andrade-Cetto[2]
[1] *Institut de Robòtica i Informàtica Industrial, CSIC-UPC*
[2] *Computer Vision Center, Universitat Autònoma de Barcelona*

Abstract

We present a framework for object detection that is invariant to object translation, scale, rotation, and to some degree, occlusion, achieving high detection rates, at 14 fps in color images and at 30 fps in gray scale images. Our approach is based on boosting over a set of simple local features. In contrast to previous approaches, and to efficiently cope with orientation changes, we propose the use of non-Gaussian steerable filters, together with a new orientation integral image for a speedy computation of local orientation.

1. Introduction

Object detection is a fundamental issue in most computer vision tasks; particularly, in applications that require object recognition. Early approaches to object recognition are based on the search for matches between geometrical object models and image features. Appearance-based object recognition gained popularity in the past two decades using dimensionality reduction techniques such as PCAs for whole-image matching. Lately, a new paradigm for object recognition has appeared based on the matching of geometrical as well as appearance local features. Moreover, the use of boosting techniques for feature selection has proven beneficial in choosing the most discriminant geometric and appearance features from training sets.

In this paper we focus on the selection of local features invariant to translation, scaling, orientation, and to some degree, occlusion. Our approach differentiates from others in that while based on boosting over a set of training samples, it can achieve object detection in real time. This is thanks to our extension of the use of steerable filters to non-Gaussian kernels, together with our proposal of a new integral image for the computation of local image orientation.

Viola and Jones [10] introduced the integral image for very fast feature evaluation. Once computed, the integral image allows the computation of Haar-like features [5] at any location or scale in real time. Unfortunately, such system is not invariant to object rotation or occlusions.

Other recognition systems that might work well in cluttered scenes are based on the computation of multi-scale local features such as the SIFT descriptor [3]. One key idea behind the SIFT descriptor is that it incorporates canonical orientation values for each keypoint. Thus, allowing scale and rotation invariance during recognition. Even when a large number of SIFT features can be computed in real time for one single image, their correct pairing between sample and test images is performed via nearest neighbor search and generalized Hough transform voting, followed by the solution of the affine relation between views; which might end up to be a time consuming process.

Yokono and Poggio [11, 12] settle for Harris corners at various levels of resolution as interest points, and from these, they select as object features those that are most robust to Gaussian derivative filters under rotation and scaling. As Gaussian derivatives are not rotation invariant, they use steerable filters [1] to steer all the features responses according to the local gradient orientation around the interest point. In the recognition phase, the system still requires local feature matching, and iterates over all matching pairs, in groups of 6, searching for the best matching homography, using RANSAC for outlier removal. Unfortunately, the time complexity or performance of their approach was not reported.

Work by many others is also related to the issue of rotation invariant feature matching [4]. We feel however, the success of our approach to be founded on the ideas presented in the former three contributions: boosting, canonical orientation, and steerable filters, along with the intro-

Financial support to M. Villamizar and A. Sanfeliu comes from the EURON Network Robot Systems Research Atelier NoE-507728, and the Spanish Ministry of Education and Science project NAVROB DPI 2004-05414. J. Andrade-Cetto is a Juan de la Cierva Postdoctoral Fellow of the Spanish Ministry of Education and Science under project TIC2003-09291, and is also funded in part by the EU PACO-PLUS project FP6-2004-IST-4-27657. The authors belong to the Artificial Vision and Intelligent Systems Group funded in part by the Catalan Research Commission. We thank Joan Purcalla for initial AdaBoost code.

0-7695-2521-0/06/$20.00 (c) 2006 IEEE

duction in this paper of the integral image for orientations, and its extension to non-Gaussian steerable filters.

In our system, keypoints are chosen as those regions in the image that have the most discriminant response under convolution with a set of wavelet basis functions at several scales and orientations. Section 2 explains how the most relevant features are selected. The selection is made with a boosting mechanism, producing a set of weak classifiers and their corresponding weights. A linear combination of these week classifiers produces a strong classifier, which is used for object detection. Rotation invariance is achieved by filtering with oriented basis functions. Filter rotation is efficiently computed with the aid of a steerable filter [1], that is, as the linear combination of basis filters, as indicated in Section 3.

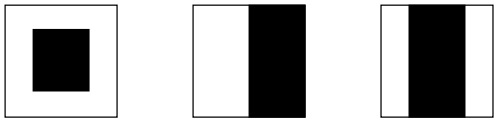
During the recognition phase, sample image regions must be rotated to a trained canonical orientation, as explained in Section 4, prior to feature matching. Such orientation is dictated by the peak on a histogram of gradient orientations, depicted in Section 5. One of the major contributions of this paper is the efficient computation of image region orientation by means of an integral image of gradient orientation histograms; enabling our system to perform object detection invariant to translation, scaling, orientation, and some degree of occlusion, in real time. Section 6 is devoted to some experimental results of the overall approach, and Section 7 has some concluding remarks.

2. Feature Selection

The set of local features that best discriminates an object is obtained by convolving positive sample images with a simplified set of wavelet basis function operators [5] at different scales and orientations. These filters have spatial orientation selectivity as well as frequency selectivity, and produce features that capture the contrast between regions representing points, edges, and strips, and have high response along for example, contours. The set of operators used is shown in Figure 1. Filter response is equivalent to the difference in intensity in the original image (or color channel magnitude) between the dark and light regions dictated by the operator.

Convolving these operators at any desired orientation is performed by steering the filter (Section 3). Furthermore, fast convolution over any region of the entire image is efficiently obtained using an integral image (Section 5).

Feature selection is performed via a boosting mechanism, namely, AdaBoost [2]. AdaBoost extracts in each iteration the weak classifier (filter width, location, type, orientation, and threshold) that best discriminates positive from negative training images. A weak classifier can be expressed as



(a) f_0 (b) f_1 (c) f_2

Figure 1. Simplified wavelet basis function set. a) center-surround b) edge, and c) line.

$$h(I) = \begin{cases} 1 & : \quad I * f > t \\ 0 & : \quad \text{otherwise} \end{cases} ,$$

where I is a training sample image, f is the filter being tested, with all its parameters (width, location, type, and orientation), $*$ indicates the convolution operation, and t is the filter response threshold. The algorithm selects the most discriminant weak classifier, as well as its contribution α in classifying the entire training set, as a function of the classification error ϵ; $\alpha = \frac{1}{2} \ln \frac{1-\epsilon}{\epsilon}$.

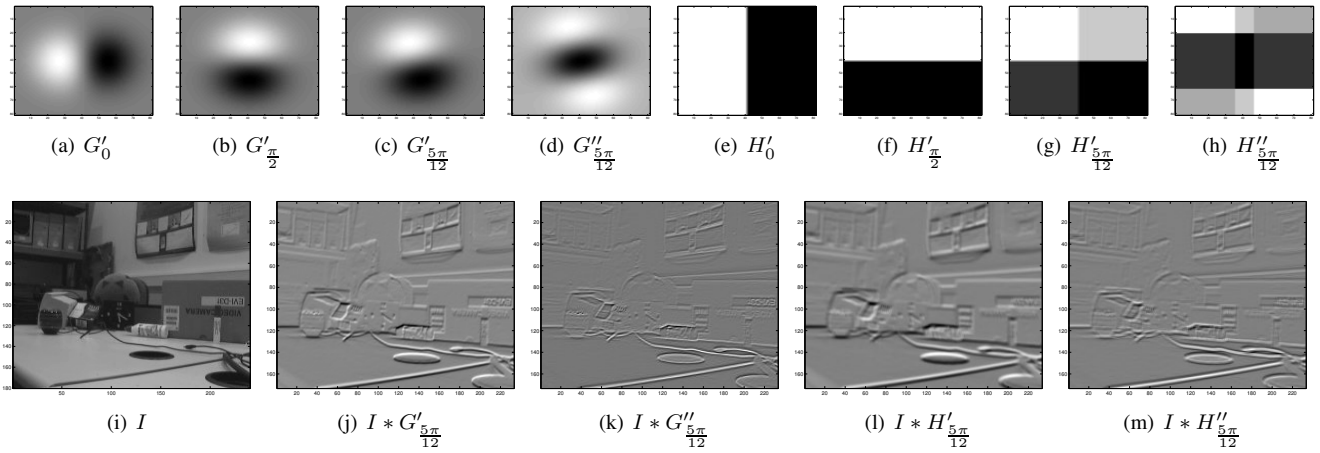
At each iteration, the algorithm also updates a set of weights over the training set. Initially, all weights are set equally, but on each round, the weights of missclassified samples are increased so that the algorithm is forced to focus on such hard samples in the training set the previously chosen classifiers missed. In a certain way, the technique is similar to a Support Vector Machine, in that both search for a class separability hyperplane, although using different distance norms, l_2 for SVMs, and l_1 for boosting [7]. The dimensionality of the separating hyperplane in AdaBoost is given by the number N of weak classifiers that form the strong classifier

$$H(I) = \begin{cases} 1 & : \quad \sum^N \alpha_i h_i(I) \geq \frac{1}{2} \sum^N \alpha_i \quad \text{object} \\ 0 & : \quad \text{otherwise} \qquad\qquad\quad \text{no-object} \end{cases} .$$

To achieve invariance to translation during the detection phase, the strong classifier H is tested for a small window the size of the training samples (30×30 pixels), and at every pixel for the entire test image. To speed up the process, the test can be performed every two or three pixels (or rows), with the compromise of possibly missing the object, i.e., having a false negative. In practice, this increment can be made up to 10% the size of the training sample, without incurring in false negatives.

Similarly, scale invariance is obtained by scaling each filter within the classifier H. Scaling of the filters can be performed in constant time for a previously computed integral image. Our tests show that we can scale up to 20% the size of the training sample, with still good detection rates.

0-7695-2521-0/06/$20.00 (c) 2006 IEEE



(a) G'_0 (b) $G'_{\frac{\pi}{2}}$ (c) $G'_{\frac{5\pi}{12}}$ (d) $G''_{\frac{5\pi}{12}}$ (e) H'_0 (f) $H'_{\frac{\pi}{2}}$ (g) $H'_{\frac{5\pi}{12}}$ (h) $H''_{\frac{5\pi}{12}}$

(i) I (j) $I * G'_{\frac{5\pi}{12}}$ (k) $I * G''_{\frac{5\pi}{12}}$ (l) $I * H'_{\frac{5\pi}{12}}$ (m) $I * H''_{\frac{5\pi}{12}}$

Figure 2. First and second order Gaussian and wavelet-based steerable filterss. (a-b) and (e-f) basis, (c-d) and (g-h) oriented filters, (i) original image, (j-m) filter responses.

3. Steerable Filters

In order to achieve orientation invariance, the local filters must be rotated previous to convolution. A good alternative is to compute these rotations with steerable filters [1], or with its complex version [8]. A steerable filter is a rotated filter comprised of a linear combination of a set of oriented basis filters, $I * f(\theta) = \sum^n k_i(\theta) I * f(\theta_i)$, where $f(\theta_i)$ are the oriented basis filters, and k_i are the coefficients of the bases.

Consider for example, the Gaussian function $G(u, v) = e^{-(u^2+v^2)}$, and its first and second order derivative filters $G'_u = -2ue^{-(u^2+v^2)}$ and $G''_u = (4u^2-2)e^{-(u^2+v^2)}$. These filters can be re-oriented as a linear combination of filter bases. The size of the basis is one more than the derivative order.

Consequently. the first order derivative of our Gaussian function at any direction θ, is $G'_\theta = \cos\theta G'_u + \sin\theta G'_v$, and a steered 2nd order Gaussian filter is obtained with $G''_\theta = \sum_{i=1}^3 k_i(\theta) G''_{\theta_i}$, with $k_i(\theta) = \frac{1}{3}(1+2\cos(\theta-\theta_i))$; and G''_{θ_i} the precomputed second order derivative kernels at $\theta_1 = 0$, $\theta_2 = \frac{\pi}{3}$, and $\theta_3 = \frac{2\pi}{3}$.

Convolving with Gaussian kernels is a time consuming process. Instead, we propose to approximate such filter response by convolving with the Haar basis from Figure 1. This, with the aid of an integral image. $I * f_1(\theta) = \cos\theta I * f_1(0) + \sin\theta I * f_1(\frac{\pi}{2})$. Similarly, filtering with our line detector at any orientation θ is obtained with $I * f_2(\theta) = \sum_{i=1}^3 k_i(\theta) I * f_2(\theta_i)$.

The similarity of the response to Haar filters allows us to use this basis instead as weak classifiers for the detection of points, edges, and lines; just as the Gaussian filters do. The main benefit of the approach is in speed of computation. While convolution with a Gaussian kernel takes time O(n) the size of the kernel, convolution with the oriented Haar basis can be computed in constant time using an integral image representation. Figure 2 shows the results of the proposed feature selection process.

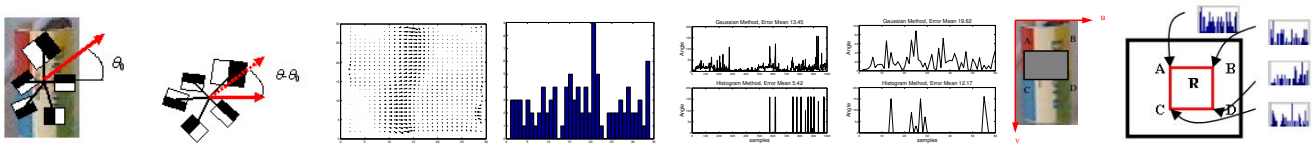
4. Local Orientation

Say, a training session has produced a constellation H of local features h as the one shown in Figure 4. Now, the objective is to test for multiple positions and scales in each new image, whether such constellation passes the test H or not. Instead of trying every possible orientation of our constellation, we chose to store the canonical orientation θ_0 of H from a reference training image block, and to compare it with the orientation θ of each image block being tested. The difference between the two indicates the amount we must re-orient the entire feature set before the test H is performed.

$$\psi = \begin{cases} \theta - \theta_0 & : \quad \theta \geq \theta_0 \\ \theta - \theta_0 + 2\pi & : \quad \text{otherwise} \end{cases}$$

On way to compute block image orientation is with ratio of first derivative Gaussians G'_u and G'_v [12], $\tan\theta = \frac{I * G'_v}{I * G'_u}$.

Another technique, more robust to partial occlusions, is to use the mode of the local gradient orientation histogram (see Figure 4), for which it is necessary to compute gradient orientations pixel by pixel, instead of a region convolution as in the previous case. When the scene is highly structured, such histogram can easily be multimodal. We follow for such cases the same convention as with SIFT features: for any peak in the histogram greater than 80% the size of the



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 3. Image orientation computed as the mode of the gradient orientation image. a) canonical orientation, b) rotated constellation, c) image gradients, d) gradient orientation histogram, local orientation error subject to e) scale change, and f) small occlusions, g) integral image, and h) local histogram integral image.

mode, a new weak classifier, oriented at that value is added to the classifier set.

We have done several tests to estimate which of these two techniques for computing local image orientation is most suitable to our needs. As shown in Figure 3(e), computing local orientation using the histogram deteriorates more with scale changes than computing the gradient over the entire image block. However, as seen in Figure 3(f), given that the mode is a nonlinear filter, the technique is much more reliable in the presence of small occlusions. We settle for the histogram mode to handle occlusions, and let the boosting mechanism deal with translation and scale affinities.

5. The Local Orientation Integral Image

An integral image is a representation of the image that allows a fast computation of features because it does not work directly with the original image intensities (color values). Instead, it works over an incrementally built image that adds feature values along rows and columns. Once computed this image representation, any one of the local features (weak classifiers) can be computed at any location and scale in constant time.

In its most simple form, the value of the integral image M at coordinates u, v contains the sum of pixels values above and to the left of u, v, inclusive, $M(u,v) = \sum_{i \leq u, j \leq v} I(i,j)$,

Then, it is possible to compute for example, the sum of intensity values in a rectangular region simply by adding and subtracting the cumulative intensities at its four corners in the integral image, Area $= A + D - B - C$.

Furthermore, the construction of the integral image is $O(n)$ in the size of the image, and is computed iteratively with $M(u,v) = I(u,v) + M(u-1,v) + M(u,v-1) - M(u-1,v-1)$.

In this form, the response from the two orthogonal Haar-filter basis from Figure 2, at any size or location, can be computed by simple adding and subtracting four values from the integral image. This, in constant time.

Extending the idea of having cumulative data at each pixel in the Integral Image, we decide to store in it orientation histogram data instead of intensity sums. Once constructed this orientation integral image, it is possible to compute a local orientation histogram for any given rectangular area within an image in constant time. Histogram(Area) $=$ Histogram(A) $+$ Histogram(D) $-$ Histogram(B) $-$ Histogram(C).

6. Experiments

For the experiments reported here, our training set had 5250 negative images and 1100 positive images. Negative images were obtained under varying illumination conditions, both from exterior and interior scenes. In order to have the boosting mechanism choose the most invariant classifiers, we have added as positive samples, synthetic images where the object to be learned appears translated, rotated, and scaled. Object translations reach 5 pixels in all directions. Scaling of the object images goes up to 20% of the original image size, and rotated images reach 10 degrees in order to aid the histogram method which was chosen to have a precision of 10 degrees, given that has 36 bins. Some positive and negative samples are shown in Figure 4.

Figure 5 shows some frames of a sequence in which the trained object is being recognized. At some point, the object is being detected at multiple neighboring locations, fact indicated by the repetitive superimposed squares. Frame (a) shows the object being detected as trained; frames (b-d) show robustness to orientation changes; frame (e) shows detection at a different scale; frames (f) and (g) show detection at both different scale and orientation; and frame (h) shows positive detection under scale, orientation, and mild occlusion.

Note however, that while convolution with the two orthogonal basis required for the first order Haar filter can be computed using an integral image; the same is not true for the second order filter since it requires basis kernels oriented at $\frac{\pi}{3}$ rad. and $\frac{2\pi}{3}$ rad., besides the already orthogonal basis

Figure 4. Positive and negative samples.

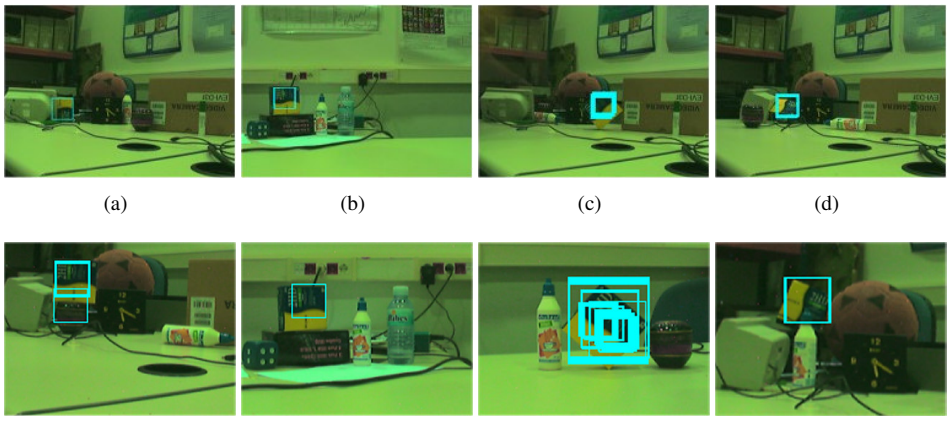


Figure 5. Some frames that show the object being detected under varying scales, orientation, and mild occlusion.

at 0 rad. Fortunately, our experiments indicate that line features are seldom chosen by the boosting algorithm as weak classifiers, accounting in the worst cases for at most 20% the total number of weak classifiers, and in little detriment of speed of computation. Nevertheless, the computation of these basis kernels in a fast integral-image-like manner is a subject of further study.

7. Conclusions

In this paper we have presented a system for object detection that is invariant to object translation, scale, rotation, and to some degree, occlusion, achieving high detection rates, at 14 fps in color images and at 30 fps in gray scale images. Our approach is based on boosting over a set of simple local features. In contrast to previous approaches, and to efficiently cope with orientation changes, we propose the use of Haar basis functions and a new orientation integral image for a speedy computation of local orientation.

References

[1] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE T. PAMI*, 13(9):891–906, 1991.

[2] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, 55(1):119–139, Aug. 1997.

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

[4] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE T. PAMI*, 27(10):1615–1630, Oct. 2005.

[5] C. P. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *Proc. IEEE ICCV*, page 555, Bombay, Jan. 1998.

[6] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE T. PAMI*, 20(1):23–38, 1998.

[7] G. Rätsch, B. Schölkopf, S. Mika, and K.-R. Müller. SVM and Boosting: One class. Tech. Rep., GMD First, Nov. 2000.

[8] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets, or "How do I organize my holiday snaps?". In *Proc. ECCV*, pages 414–431, Copenhagen, 2002.

[9] B. Schiele and J. L. Crowley. Object recognition using multidimensional receptive field histograms. In *Proc. ECCV*, pages 610–619, Cambridge, Apr. 1996.

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

[11] J. Yokono and T. Poggio. Oriented filters for object recognition: An empirical study. In *Proc. 6th IEEE Int. Conf. Automatic Face Gesture Recog.*, pages 755–760, Seoul, 2004.

[12] J. Yokono and T. Poggio. Rotation invariant object recognition from one training example. Tech. Rep. 2004-010, MIT AI Lab., Apr. 2004.