

Receptive Field Cooccurrence Histograms for Object Detection

Staffan Ekvall

Computational Vision and Active Perception
Royal Institute of Technology, Stockholm, Sweden
ekvall@nada.kth.se

Danica Kragić

Centre for Autonomous Systems
Royal Institute of Technology, Stockholm, Sweden
danik@nada.kth.se

Abstract—Object recognition is one of the major research topics in the field of computer vision. In robotics, there is often a need for a system that can locate certain objects in the environment - the capability which we denote as 'object detection'. In this paper, we present a new method for object detection. The method is especially suitable for detecting objects in natural scenes, as it is able to cope with problems such as complex background, varying illumination and object occlusion. The proposed method uses the receptive field representation where each pixel in the image is represented by a combination of its color and response to different filters. Thus, the cooccurrence of certain filter responses within a specific radius in the image serves as information basis for building the representation of the object.

The specific goal in this work is the development of an on-line learning scheme that is effective after just one training example but still has the ability to improve its performance with more time and new examples. We describe the details behind the algorithm and demonstrate its strength with an extensive experimental evaluation.

I. INTRODUCTION

Recognizing objects is a central challenge in the field of computer vision and autonomous robotics. Although a significant amount of work has been reported, the proposed methods still differ significantly regarding these two research areas. In this paper, we consider an autonomous robot scenario. Here, the robots are to operate in complex home and office environments which means that it is very difficult to model all possible objects that the robot is supposed to manipulate. In addition, the object recognition method used has to be robust to outliers and changes commonly occurring in such a dynamic environment.

In terms of object recognition, the appearance based representations are commonly used, [1]–[3]. However, the appearance based methods suffer from various problems. For example, a representation based on the color of an object is sensitive to varying lighting conditions, while a representation based on the shape of an object is sensitive to occlusion. In this paper, we present an approach based on Receptive Field Cooccurrence Histograms, that robustly copes with both of the mentioned problems. In most of the recognition methods reported in the literature, a large number of training images are needed to recognize objects viewed from arbitrary angles. The training is often performed off-line and, for some algorithms, it can be very time consuming. For robotic applications, it is important that new objects can be learned easily. i.e. putting

a new object in the database and retraining should be fast and computationally cheap.

Our goal in the work reported in this paper is to develop an on-line learning scheme than can be effective after just one training example but still has the ability to improve its performance with more examples. Also, learning new objects should be possible without heavy recalculations on already learned objects.

The vision system described in this paper is a part of an mobile autonomous robot system that, among others, performs pick-and-place tasks. In addition, we intend to use the algorithm in a programming by demonstration framework, for automatic interpretation of the teacher's instructions, [4]. We believe the method is general enough to be easily used for a variety of other applications requiring robust object detection.

A. Object Detection vs. Object Recognition

An object recognition algorithm is typically designed to classify an object to one of a several predefined classes assuming that the segmentation of the object has already been performed. Commonly, the test images show a single object that is centered in the image and occupies most of the image area. The test image may also have a black background [5], making the task even more simple.

The task for an object detection algorithm is much harder. Its purpose is to search for a specific object in an image of a complex scene. Most of the object recognition algorithms may be used for object detection by using a search window and scanning the image for the object. Regarding the computational complexity, some methods are more suitable for searching than others.

In this work, we present a recognition algorithm that performs very well when used both for detection and recognition. Despite a cluttered background and occlusion, it is able to detect the specific object among several other similar looking objects. This property makes the algorithm ideal for use on robotic platforms which are to operate in natural scenes.

This paper is organized as follows: in Section II we briefly summarize some of the related work in object recognition and detection. In Section III we describe the Receptive Field Cooccurrence Histogram. Its use for object detection is explained in Section IV, and the detection algorithm is then evaluated in Section V. Finally, Section VI concludes this paper.

II. RELATED WORK

Back in 1991, Swain and Ballard [6] demonstrated how RGB color histograms can be used for object recognition. Schiele *et al.* [7] generalized this idea to histograms of receptive fields, and computed histograms of either first-order Gaussian derivative operators or the gradient magnitude and the Laplacian operator at three scales. In [8], Linde *et al.* evaluated more complex descriptor combinations, forming histograms of up to 14 dimensions. Excellent performance on both the COIL-100 and the ETH-80 database was shown. Mel [9] also developed a histogram based object recognition system that uses multiple low-level attributes such as color, local shape and texture. Although these methods are robust to changes in rotation, position and deformation, they cannot cope with recognition in a cluttered scene. The problem is that the background visible around the object confuses the methods.

In [10], Chang *et al.* shows how color cooccurrence histograms can be used for object detection, performing better than regular color histograms. We have further evaluated the color cooccurrence histograms. In [11], we use them for both object detection and pose estimation.

The methods mentioned so far are *global* methods, meaning that they calculate the object representation on all available image data. In contrast, local feature-based methods only capture the most representative parts of an object. In [12], Lowe presents the SIFT features, which is a promising approach for detecting objects in natural scenes. However, the method relies on the presence of feature points and, for objects with simple or no texture, this method fails.

Detecting human faces is another area of object detection. In [13], Viola *et al.* detects human faces using an algorithm based on the occurrence of simple features. Several weak classifiers are integrated through boosting, and the final classifier is able to detect faces in natural, cluttered scenes although a number of false positives cannot be avoided. However, it is unclear whether the method can be used to detect arbitrary objects or handle occlusion. Detecting faces can be regarded as a problem of detecting a category of objects in contrast to this work where we deal with the problem of detecting a specific object.

III. RECEPTIVE FIELD COOCCURRENCE HISTOGRAM

A Receptive Field Histogram is a statistical representation of the occurrence of several descriptor responses within an image. Examples of such image descriptors are color intensity, gradient magnitude and Laplace response, described in detail in Section III-A. If only color descriptors are taken into account, we have a regular color histogram.

A Receptive Field Cooccurrence Histogram (RFCH) is able to capture more of the geometric properties of an object. Instead of just counting the descriptor responses for each pixel, the histogram is built from *pairs* of descriptor responses. The pixel pairs can be constrained based on, for example, their relative distance. This way, only pixel pairs separated by less than a maximum distance, d_{max} are considered. Thus, the histogram represents not only how common a certain

descriptor response is in the image but also how common it is that certain combinations of descriptor responses occur close to each other.

A. Image Descriptors

We will evaluate the performance of histogram based object detection using different types of image descriptors. The descriptors we use are all rotationally and translationally invariant. If rotational invariance is not required for a particular application, increased recognition rate could be achieved by using for example Gabor filters. In brief, we will consider the following basic types of image descriptors, as well as various combinations of these:

- **Normalized Colors**

The color descriptors are the intensity values in the red and green color channels, in normalized RG-color space, according to $r_{norm} = \frac{r}{r+g+b}$ and $g_{norm} = \frac{g}{r+g+b}$.

- **Gradient Magnitude**

The gradient magnitude is a differential invariant, and is described by the combination of partial derivatives (L_x, L_y): $|\nabla L| = \sqrt{L_x^2 + L_y^2}$. The partial derivatives are calculated from the scale-space representation $L = g * f$ obtained by smoothing the original image f with a Gaussian kernel g , with standard deviation σ .

- **Laplacian**

The Laplacian is an on-center/off-surround descriptor. Using this descriptor is biologically motivated, as it is well known that center/surround ganglion cells exist in the human brain. The Laplacian is calculated from the partial derivatives (L_{xx}, L_{yy}) according to $\nabla^2 L = L_{xx} + L_{yy}$. From now on, $\nabla^2 L$ denotes calculating the Laplacian on the intensity channel, while $\nabla^2 L_{rg}$ denotes calculating it on the normalized color channels separately.

B. Image Quantization

Regular multidimensional receptive field histograms [7] have one dimension for each image descriptor. This makes the histograms huge. For example, using 15 bins in a 6-dimensional histogram means 15^6 ($\sim 10^7$) bin entries. As a result the histograms are very sparse, and most of the bins have zero or only one count. Building a cooccurrence histogram makes things even worse, in that case we need about 10^{14} bin entries. By first clustering the input data, a dimension reduction is achieved. Hence, by choosing the number of clusters, the histogram size may be controlled. In this work, we have used 80 clusters resulting in that our cooccurrence histograms are dense and most bins have high counts.

Dimension reduction is done using K-means clustering [14]. Each pixel is quantized to one of N cluster centers. The cluster centers have a dimensionality equal to the number of image descriptors used. For example, if both color, gradient magnitude and the Laplacian are used, the dimensionality is six (three descriptors on two colors). As distance measure, we use the Euclidean distance in the descriptor space. That is, each cluster has the shape of a sphere. This requires all input dimensions to be of the same scale, otherwise some

descriptors would be favored. Thus, we scale all descriptors to the interval $[0,255]$. The clusters are randomly initialized, and a cluster without members is relocated just next to the cluster with the highest total distance over all its members. After a few iterations, this leads to a shared representation of that data between the two clusters. Each object ends up with its own cluster scheme in addition to the RFCH calculated on the quantized training image.

When searching for an object in a scene, the image is quantized with the same cluster-centers as the cluster scheme of the object being searched for. Quantizing the search image also has a positive effect on object detection performance. Pixels lying too far from any cluster in the descriptor space are classified as the background and not incorporated in the histogram. This is because each cluster center has a radius that depends on the average distance to that cluster center. More specifically, if a pixel has a Euclidean distance d to a cluster center, it is not counted if $d > \alpha \cdot d_{avg}$, where d_{avg} is the average distance of all pixels belonging to that cluster center (found during training), and α is a free parameter. We have used $\alpha = 1.5$ i.e., most of the training data is captured. $\alpha = 1.0$ corresponds to capturing about half the training data.

Fig. 1 shows an example of a quantized search image, when searching for a red, green and white Santa-cup.



Fig. 1. Example when searching for the Santa-cup, visible in the top right corner. Left: The original image. Right: Pixels that survive the cluster assignment. The pixels that lie too far away from their nearest cluster are ignored (set to black in this example). The red striped table cloth still remains, as the Santa cup contains red-white edges.

The quantizing of the image can be seen as a first step that simplifies the detection task. To maximize detection rate, each object should have its own cluster scheme. This, however, makes it necessary to quantize the image once for each object being searched for. If several different objects are to be detected and a very fast algorithm is required, it is better to use shared cluster centers over all objects known. In that case, the image only has to be quantized once.

It has to be noted that multiple histograms of the object across a number of training images may share the same set of cluster centers.

C. Histogram Matching

The similarity between two normalized RFCHs is computed as the histogram intersection:

$$\mu(h_1, h_2) = \sum_{n=1}^N \min(h_1[n], h_2[n]) \quad (1)$$

where $h_i[n]$ denotes the frequency of receptive field combinations in bin n for image i , quantized into N cluster centers. The higher the value of $\mu(h_1, h_2)$, the better the match between the histograms. Prior to matching, the histograms are normalized with the total number of pixel pairs.

Another popular histogram similarity measure is the χ^2 :

$$\mu(h_1, h_2) = \sum_{n=1}^N \frac{(h_1[n] - h_2[n])^2}{h_1[n] + h_2[n]} \quad (2)$$

In this case, the lower value of $\mu(h_1, h_2)$, the better the match between the histograms. The χ^2 similarity measure usually performs better than the histogram intersection method on object recognition image databases. However, we have found that χ^2 performs much worse than histogram intersection when used for object detection. We believe that this is because the background that is visible in the search window and not present during training, is not penalizing the match correspondence as much as with the χ^2 . Histogram intersection focuses on bins that represent the searched object best, while χ^2 treats all bins equally. As mentioned, χ^2 still performs slightly better on object recognition databases. In these databases there is often only a black background, or even worse, the background provides information about the object (e.g., airplanes shown on a blue sky background).

IV. OBJECT DETECTION USING RFCHS

Object detection is a more challenging task than object recognition. Usually, the object only occupies a small area of the image and object recognition algorithms cannot be run directly considering the entire image. Instead, the image is scanned using a small search window. The window is shifted such that consecutive windows overlap to 50 % and the RFCH of the window is compared with the object's RFCH according to (1). Each object may be represented by several histograms if its appearance changes significantly with the view angle of the object. However, in this work we only used one histogram per object.

The matching vote $\mu(h_{object}, h_{window})$ indicates the likelihood that the window contains the object. Once the entire image has been searched through, a vote matrix provides a hypothesis of the object's location. Fig. 2 shows a typical scene from our experiments together with the corresponding vote matrix for the yellow soda can. The vote matrix reveals a strong response in the vicinity of the object's true position close to the center of the image.

The vote matrix may then be used to segment the object from the background, as described in [11], or just provide an hypothesis of the object's location. The most probable location is corresponding to the vote cell with the maximum value.

A. Complexity

The running time can be divided into three parts. First, the test image is quantized. The quantization time grows linearly with N . Second, the histograms are calculated. The calculation time grows with the square of d_{max} . Last, the histogram



Fig. 2. Example of searching for the yellow soda can, placed closed to the center of the image. Dark areas indicate high likelihood of the object being present. There is a strong response where the soda can is placed, but also a small response at the location of the raisin box, standing to the left in the image, because of the similar yellow color.

similarities are calculated. Although histogram matching is a fast process, its running time grows with the square of N .

The algorithm is very fast which makes it applicable even on mobile robots. Depending on the number of descriptors used and the image size, the algorithm implemented in C++ runs at about 3-10 Hz on a 3 GHz regular PC.

V. EXPERIMENTAL EVALUATION

We evaluate six different descriptor combinations in this section. The descriptor combinations are chosen to show the effect of the individual descriptors as well as the combined performance. The descriptor combinations are:

- $[R, G]$ - Capturing only the absolute values of the normalized red and green channel. Corresponding to a color cooccurrence histogram. With $d_{max} = 0$ this means a normal color histogram (except that the colors are clustered).
- $[R, G, \nabla^2 L_{rg}, \sigma = 2]$ - The above combination extended with the Laplacian operator at scale $\sigma = 2$. As the operator works on both color channels independently, this combination has dimension 4.
- $[R, G, |\nabla L_{rg}|, \nabla^2 L_{rg}, \sigma = 2]$ - The above combination extended with the gradient magnitude information on each color channel, scale $\sigma = 2$.
- $[|\nabla L|, \sigma = 1, 2, 4, |\nabla L_{rg}|, \sigma = 2]$ - Only the gradient magnitude, on the intensity channel and on each color channel individually. On the intensity channel, three scales are used, $\sigma = 1, 2, 4$. For each of the color channels, scale $\sigma = 2$ is used. 5 dimensions.
- $[\nabla^2 L, \sigma = 1, 2, 4, \nabla^2 L_{rg}, \sigma = 2]$ - The same combination as above, but for the Laplacian operator instead.
- $[R, G, |\nabla L_{rg}|, \nabla^2 L_{rg}, \sigma = 2, 4]$ - The combination of colors, gradient magnitude and the Laplacian, on two different scales, $\sigma = 2, 4$. 10 dimensions.

All descriptor combinations were evaluated using CODID - CVAP Object Detection Image Database, [15].

A. CODID - CVAP Object Detection Image Database

CODID is an image database designed specifically for testing object detection algorithms in a natural environment. The database contains 40 test images of size 320x200 pixels, and each image contains 14 objects. The test images include problems such as object occlusion, varying illumination and



Fig. 3. The ten images used for training.

textured background. Out of the 14 objects, 10 are to be detected by an object detection algorithm. The database provides 10 training images for this purpose, i.e. only one training image per object. The database also provides bounding boxes for each of the ten objects and each scene and an object is considered to be detected if the algorithm can provide pixel coordinates within the object's bounding box for that scene. In general, detection algorithms may provide several hypotheses of an object's location. In this work, only the strongest hypothesis is taken into account.

The test images are very hard from a computer vision point of view, with cluttered scenes and objects lying rotated behind and on top of each other. Thus, many objects are partially occluded in the scene. In total, the objects are arranged in 20 different ways and each scene is captured under two lighting conditions. The first lighting condition is the same as during training, a fluorescent ceiling lamp, while the second is a closer placed light bulb illuminating from a different angle.

B. Training

For training, one image of each object is provided. Naturally, providing more images would improve the recognition rate but our main interest is to evaluate the proposed method using just one training image. The training images are shown in Fig. 3. As it can be seen, some objects are very similar to each other, making the recognition task non-trivial. The histogram is built only from non-black pixels. In these experiments, the training images have been manually segmented. For training in robotic applications, we assume that the robot can observe the table before and after the object is placed in front of the camera and perform the segmentation based on image differencing.

C. Detection Results

The experimental evaluation has been performed using six combinations of feature descriptors. As it can be seen in Table I, the combination of all feature descriptors gives the best results. The color descriptor is very sensitive to changing lighting conditions, despite the fact that the images were color normalized prior to recognition. On the other hand, the other descriptors are very robust with respect to this problem and the combination of descriptors performs very well. We also compare the method with regular color histograms which show much worse results.

Adding descriptors on several scales does not seem to improve the performance in the first case, but when lighting conditions change, some improvement can be seen. With changed illumination, colors are less reliable and the method is able to benefit from the extra information given by the Laplace and gradient magnitude descriptors on several scales. All descriptor combinations have been tested with $N = 80$ cluster-centers, except the 10-dimensional one which required 130 cluster-centers to perform optimally. Also, $d_{max} = 10$ was used in all tests, except for the color histogram method, which of course use $d_{max} = 0$.

All detection rates reported in Table I are achieved using the histogram intersection method (1). For comparison, we also tested the 6D descriptor combination with the χ^2 method (2). With this method, only 60 % of the objects were detected, compared to 95 % using histogram intersection.

TABLE I

THE DETECTION RATE OF DIFFERENT FEATURE DESCRIPTOR COMBINATIONS IN TWO CASES: I) SAME LIGHTING CONDITIONS AS WHEN TRAINING, AND II) CHANGED LIGHTING CONDITIONS.

Descriptor Combination:	Lighting Condition: Same	Lighting Condition: Changed
2D: Color histogram	71.5	38.0
2D: $[R, G]$ (CCH)	77.5	38.0
4D: $[R, G, \nabla^2 L_{rg}, \sigma = 2]$	88.5	61.5
5D: $[\nabla L , \sigma = 1, 2, 4, \nabla L_{rg} , \sigma = 2]$	57	51
5D: $[\nabla^2 L, \sigma = 1, 2, 4, \nabla^2 L_{rg}, \sigma = 2]$	77.5	62.0
6D: $[R, G, \nabla L_{rg} , \nabla^2 L_{rg}, \sigma = 2]$	95.0	80.0
10D: $[R, G, \nabla L_{rg} , \nabla^2 L_{rg}, \sigma = 2, 4]$	93.5	86.0

1) *Misclassification Analysis:* Among the objects to be detected, there are three quite similar cups as shown in Fig 3. Most detection errors originate from these three cups being confused with each other. If one cup is partially occluded, its appearance may be less similar to its training image compared to the appearance of the other cups. Thus, in these cases, the algorithm suggest the location of another cup as the most probable location.

D. Free parameters

The algorithm requires setting a number of parameters which were experimentally determined. However, it is shown that the detection result are not significantly affected by the values of parameters. The parameters are:

- **Number of cluster-centers, N**

We found that using too few cluster-centers reduces the detection rate. From Fig. 4 it can be seen that feature descriptor combinations with high dimensionality require more cluster centers to reach their optimal performance. As seen, 80 clusters is sufficient for most descriptor combinations.

- **Maximum pixel distance, d_{max}**

The effect of cooccurrence information is evaluated by varying d_{max} . Using $d_{max} = 0$ means no cooccurrence information. As seen in Fig. 5, the performance is increased radically by just adding the cooccurrence information of

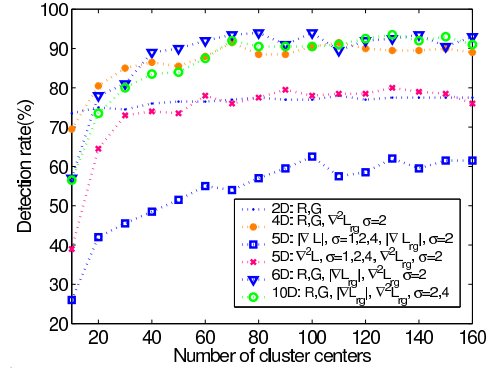


Fig. 4. The importance of the number of cluster-centers for different image descriptor combinations.

pixel neighbors, $d_{max} = 1$. For $d_{max} > 10$ the detection rate starts to decrease. This can be explained by the fact that the distance between the pixels is not stored in the RFCH. Using a too large maximum pixel distance will add more noise than information, as the likelihood of observing the same cooccurrence in another image decreases with pixel distance. As seen in Fig. 5, the effect of the cooccurrence information is even more significant when lighting conditions change.

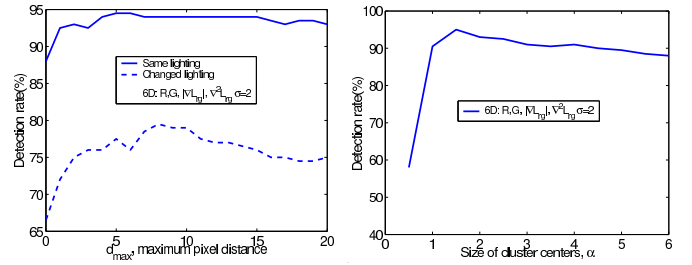


Fig. 5. Left: The detection rate mapped to the maximum pixel distance used for cooccurrence, d_{max} , in two cases: Same lighting as when training, and different lighting from training. Right: The detection rate mapped to the size of the cluster centers, α .

- **Size of cluster-centers, α**

We have investigated the effect of limiting the size of the cluster centers. Pixels that lie outside all of the cluster centers are classified as background and not taken into account. As seen in Fig. 5, the algorithm performs optimally when $\alpha = 1.5$, i.e. the size is 1.5 times the average distance to the cluster center used during training. Smaller α removes too many of the pixels and, as α grows, the effect described in Section III-B starts to decrease.

- **Search window size**

We have found that some object recognition algorithms require a search window size of a specific size to function properly for object detection. This is a serious drawback, as the proper search window size is commonly not known in advance. Searching the image several times with different sized search windows is a solution, although it

is quite time consuming. As Fig. 6 shows, the choice of search window size is not crucial for the performance of our algorithm. The algorithm performs equally well for window sizes of 20 to 60 pixels. In our experiments, we have used a search window of size 40.

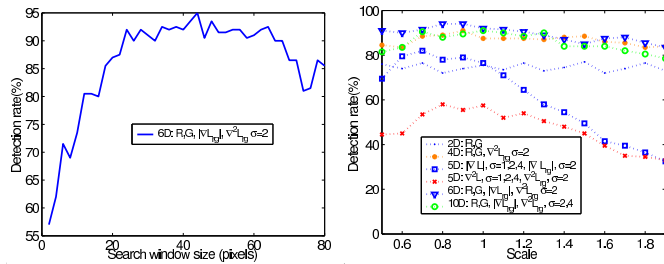


Fig. 6. Left: The detection rate mapped to the size of the search window. Right: The effect on detection rate when the training examples are scaled, for six different image descriptor combinations.

E. Scale Robustness

We have also investigated how the different descriptor combinations performs when the scale of the training object is changed. The training images were rescaled to between half size and double size, and the effect on detection performance was investigated. As seen in Fig. 6, the color descriptors are very robust to scaling, while the other descriptors types decrease in performance as the scale increase. However, when the descriptor types are combined, the performance is partially robust to scale changes. To improve scale robustness, the image can be scanned at different scales.

VI. CONCLUSION

In this paper, a new method for object detection has been presented. For representing an object, Receptive Field Cooccurrence Histograms are used. A Receptive Field Histogram represents how common certain filter responses and colors are in an image. Then, a Receptive Field Cooccurrence Histogram, RFCH, is a representation of how often pairs of certain filter responses and colors lie close to each other in the image. This means that more geometric information is preserved and the recognition task becomes easier. The experimental evaluation shows that the method is able to successfully detect objects in cluttered scenes by comparing the histogram of a search window with the stored representation.

We have shown that the method is very robust. The representation is invariant to translation and rotation and robust to scale changes and illumination variations. The algorithm is able to detect and recognize many objects with different appearance, despite severe occlusions and cluttered backgrounds. The performance of the method depends on a number of parameters but we have shown that the choice of these are not crucial. On the contrary, the algorithm performs very well with a wide variety of parameter values.

The strength in the algorithm lies in its applicability to object detection for robotic applications. There are several object recognition algorithms that perform very well on object

recognition image databases assuming that the object is centered in the image on a uniform background. For an algorithm to be used for object detection, it has to be able to recognize the object although it is placed on a textured cloth and only partially visible. The CODID image database was specifically designed for testing these types of natural challenges, and we have reported good detection results on this database. The algorithm is fast and fairly easy to implement. Training of new objects is a simple procedure and only a few images are sufficient for a good representation of the object.

There is still place for improvement. The cluster-center representation of the descriptor values is not ideal, and more complex quantization methods are to be investigated. In the experiments we recognized only 10 objects. More experiments are required to evaluate how the algorithm scales with an increasing number of objects, and also to investigate the method's capability to generalize over a class of objects, for example *cups*. In this work, we have used three types of image descriptors considered on several scales, but there is no upper limit of how many descriptors the algorithm may handle. There may be other types of descriptors that would improve results, and additional types of descriptors will be considered in the future.

REFERENCES

- [1] H. Murase and S. K. Nayar, "Visual learning and recognition of 3-d objects from appearance," *International Journal of Computer Vision*, vol. 14, pp. 5–24, 1995.
- [2] A. Selinger and R. C. Nelson, "Appearance-based object recognition using multiple views," Tech. Rep. 749, Comp. Sci. Dept. University of Rochester, Rochester NY, June 2001.
- [3] B. Caputo, *A new kernel method for appearance-based object recognition: spin glass-Markov random fields*. PhD thesis, Royal Institute of Technology, Sweden, 2004.
- [4] S. Ekvall and D. Kragic, "Grasp recognition for programming by demonstration," in *IEEE/RSJ International Conference on Robotics and Automation, ICRA'05*, 2005.
- [5] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library: Coil-100," in *Technical Report CUCS-006-96, Department of Computer Science, Columbia University*, 1996.
- [6] M. Swain and D. Ballard, "Color indexing," *IJCV7*, pp. 11–32, 1991.
- [7] B. Schiele and J. L. Crowley, "Recognition without correspondence using multidimensional receptive field histograms," *International Journal of Computer Vision*, vol. 36, no. 1, pp. 31–50, 2000.
- [8] O. Linde and T. Lindeberg, "Object recognition using composed receptive field histograms of higher dimensionality," in *17th International Conference on Pattern Recognition, ICPR'04*, 2004.
- [9] B. Mel, "SEEMORE: Combining Color, Shape and Texture Histogramming in a Neurally Inspired Approach to Visual Object Recognition," *Neural Computation*, vol. 9, pp. 777–804, 1997.
- [10] P. Chang and J. Krumm, "Object recognition with color cooccurrence histograms," in *CVPR'99*, pp. 498–504, 1999.
- [11] S. Ekvall, F. Hoffmann, and D. Kragic, "Object recognition and pose estimation for robotic manipulation using color cooccurrence histograms," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'03*, 2003.
- [12] D. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision*, pp. 1150–1157, 1999.
- [13] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *CVPR*, 2001.
- [14] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," in *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 1:281–297, University of California Press, 1967.
- [15] "CODID - CVAP Object Detection Image Database." <http://www.nada.kth.se/~ekvall/codid.html>.