Object state recognition for automatic AR-based maintenance guidance

Pavel Dvorak Radovan Josth Elisabetta Delponte

Konica Minolta Laboratory Europe
research.konicaminolta.eu

Abstract

This paper describes a component of an Augmented Reality (AR) based system focused on supporting workers in manufacturing and maintenance industry. Particularly, it describes a component responsible for verification of performed steps. Correct handling is crucial in both manufacturing and maintenance industries and deviations may cause problems in later stages of the production and assembly. The primary aim of such support systems is making the training of new employees faster and more efficient and reducing the error rate. We present a method for automatically recognizing an object's state with the objective of verifying a set of tasks performed by a user. The novelty of our approach is that the system can automatically recognize the state of the object and provide immediate feedback to the operator using an AR visualization enabling fully automatic step-by-step instructions.

1. Introduction

Augmented Reality (AR) is an emerging technology and one of the expected solutions used in Industry 4.0, the current trend of automation and digitization in manufacturing industry. The main use case of such technology is supporting the workers during their work, mainly tasks that are not done by worker on a daily basis. Another big group of target users are new employees. The main benefit of making the training of new employees faster and more efficient is related to cost savings. There is a big potential for smart glasses, another emerging technology tightly connected to AR, especially in industry.

The problem of automatically detecting and recognizing objects from visual information has been extensively studied [11] [7]. Although in several applications the results achieved are satisfactory, there are many cases for which improved solutions still remain to be identified. For instance, in several applications concerned with monitoring of tasks in manufacturing and maintenance processes in industry, the targeted objects may have highly similar appearances, the background of the object may be cluttered and occlusions may frequently occur.

In the present paper, a method is proposed that is adapted to the task of automatically recognizing an object's state within the environment of an industrial manufacturing or maintenance process with the objective of verifying a set of tasks performed by an operator using an Augmented Reality (AR). By exploiting a cyber physical system, in this case a pair of smart glasses that integrates a Holographic Optical Element (HOE) [4], the method presented in the current paper is able to support the operator to successfully complete maintenance or manufacturing activities in an industrial context.

The combination of the smart glasses and object state recognition method can inform the user in the case of incorrect handling of objects or other instruments and provide relevant information concerning the performed task. A further extension of this method includes aiding the navigation of the operator through the maintenance and manufacturing workflow to improve the process efficiency and reduce human errors. In addition, users operating in harsh environment may be benefit from the smart glasses interface to more easily obtain access to data presented together with its context, including information about the users environmental conditions.

To monitor the range of tasks that are envisaged to be performed by the user the cyber physical system necessarily acquires images of a large number of objects, and that these objects will be quite similar in appearance. The system has to provide feedback to the operator on how well they have performed the specified actions. Within the context of the manipulation of devices necessary to perform maintenance-type activities accurate recognition of objects must be made quickly to provide the real-time support needed whilst performing the tasks.

The algorithms proposed as part of the method presented here are based on HOG [2] because of the robustness of this method to changes in light conditions, and its capability to recognize non-textured objects.

The dataset of images that has been created for the object state evaluation application comprises pictures of objects with small variations in their appearance and with a limited number of states (up to 30 instances per objects). Within this paper, the success rate of object recognition within the dataset is presented. The goal was to identify an object's subtype and therefore the dataset concentrated on 2D object images and did not deal with a large-scale visual search engine since several efficient methods have already been proposed for such a purpose as described in [12] and [5].

In this paper, we propose a solution for the augmented reality-based automated step-by-step guidance systems with application in maintenance and manufacturing industry. The benefit of such technology is an increased efficiency of a worker as well as a reduced number of errors caused by incorrect handling.

2. Methodology

The method proposed in the present paper is concerned with the recognition of similar objects (and focused only on 2D objects). It has been designed to distinguish different instances of an object that can have more than a single appearance or more than one particular state. An example of such a situation is depicted in Figure 1.

The first step in the detection of the instance is the necessity to recognize the object itself and for this purpose, any suitable method for fast object detection can be used. This can include a dictionary-based method (such as Bag-of-Words [9]) for large-scale databases or a feature-based method for small-scale databases (based simply on any type of feature descriptors such as the well-known SIFT [6], SURF [1], ORB [8], etc.). The only requirement of such a method is that it provides the coordinates and I.D. of the detected object. A more complete description of these fast object detection methods is not given here, since the present work is more concerned with the object instance detection problem.

Once the object has been detected and aligned with the template picture of the object, the next step is the sub-identification phase, i.e. the recognition of the object's instance or state. The proposed method makes use of a descriptor called Histogram of Oriented Gradients (HOG) [2]. In this section a brief summary of this well-known technique is given together with a description of both how the reference database was created (training) and how to apply the method to a new image (testing). A block diagram summarizing the method is given in Figure 2. The figure shows the complete processing pipeline. The basis of the proposed method (HOG features) as well as both parts of the application of the method (training and testing) is described below. HOG features provides a solution that is based on gradients that provide the information about the spatial and structural visual appearance of the object.

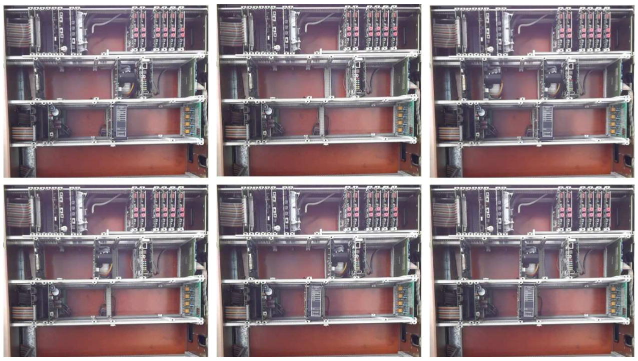

Figure 1. Example of several different states of a sample object. Sample object: a rack that needs to be filled with various types of electronic boards. State: positions occupied by boards

The requirements for the algorithm that has been developed can be stated as follows:

- The method should work in real-time.
- The method must be able to distinguish two or more similar objects.
- The method must be robust to changing light conditions, which eliminates those methods that rely on colour descriptions
- The method must work for both textured and textureless areas, which eliminates those methods based on feature points detection

2.1. HOG Descriptor

Histogram of Oriented Gradients is a well-known descriptor in Computer Vision and Image Processing, first introduced by Dalal and Triggs in [2]. Since then, several applications of these descriptors have been defined for object detection and recognition. The method is based on the number of occurrences of gradient orientations in localized portions of an image, similar to Scale-invariant feature transform [6] and the shape context. It significantly differs from the Scale-invariant feature because it is computed on a dense grid of uniformly spaced cells and it uses overlapping local contrast normalization to improve the accuracy.

The basic concept behind the HOG features descriptor is that local object appearance and shape can often be characterized by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradients or edge positions. In practice, with this approach, the image window is divided into small spatial regions (cells). A local 1-D histogram of gradient directions or edge orientations is accumulated over the pixels of each cell. The combined histogram entries form the representation of the object. To obtain improved invariance to illumination, shadowing, etc., it is necessary to contrast-normalize the local responses of the descriptor before using

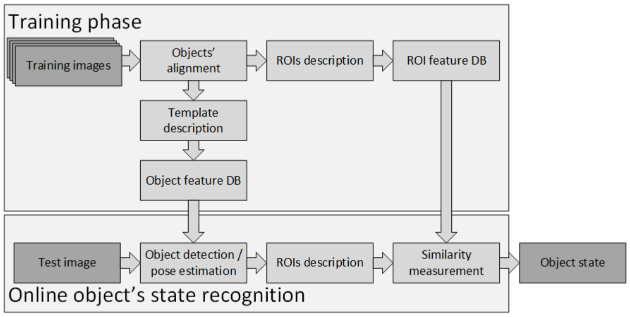

Figure 2. Block diagram summarizing theproposed methodr

them. In the present work, the normalized descriptor blocks are referred to as Histogram of Oriented Gradient (HOG) descriptors.

2.2. Reference Dataset (and Training)

Typically, the first step necessary for visual search engines or to match a scene captured by camera to provide an augmented reality experience, is to specify the database of the target objects $\mathbf{O}$ (or pictures to be indexed and searched). This same approach is also required with the method presented in this paper, however, some additional processing is required. Each object $\mathbf{o} \in \mathbf{O}$, whose state is to be determined, requires an image of each particular state s that is aligned with all other pictures of the object. The alignment is based on the same technology as the object detection during the object state recognition described in the next section. The areas that distinguish a particular object state s are then determined and are represented by a binary mask. Note that several object states may be represented by a single mask and that a mask can contain one or more areas. Generating the masks is typically a manual process, however, automated mask generating methods may be based on those used for automatic change detection described e.g. by [10]. A representative mask for an object with three states is shown in Figure 3.

Following the definition of the pictures and masks for each object, the feature database $\mathbf{F}$ for each image of a particular state s is created. The feature database is used during the recognition phase of the Histogram of Oriented Gradients (HOG) features method. A single HOG feature is computed per mask. The HOG features require that the area for the computation is rectangular and therefore the bounding rectangle of the target area is considered. Since the rectangular area can still vary in shape, i.e. its height and width, the parameters of the HOG feature, particularly the block size and stride and the cell size, should be adapted. The size of the area, equal to the window size, is adapted in order to be of a size usable for HOG descriptor. Therefore the length of the feature word may vary per mask, per state and per object, i.e. a state s may be represented by a set of code-words of various lengths.

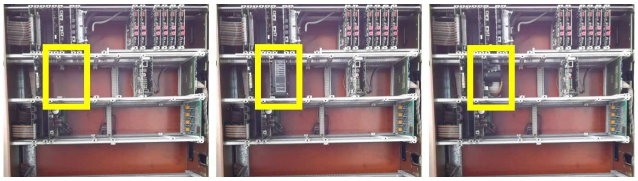

Figure 3. Example of an object with three states represented by a single mask (its outline is highlighted by the yellow rectangle).

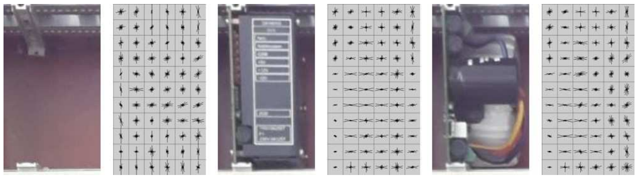

Figure 4. HOG features for three states of a sample object with a single mask.

Visualization of HOG features for a sample object with three different states is depicted in Figure 4. The same object and region as shown in Figure 3 has been used.

2.3. Object State Recognition

A prerequisite of a successful recognition of an object state or instance, is to correctly identify the visualized object in the scene as well as to estimate its pose and location as precisely as possible. Technologies employed in AR engines may be used to meet these requirements. We use the approach described by Girod et al. in [3]. Any other similar method used for visual search may be employed. The only requirement is that the method provides transformation matrix from the template to the scene space in order to correctly map the detected object to the template.

After the object o has been identified and its position determined, its state can then be recognized. The image is transformed to the space of the template so that the areas to be verified in the query image are aligned to the same location as they appear in the template image. This inverse transformation may result in some information loss but is necessary to compute the HOG features from the same rectangular areas. This condition cannot be consistently satisfied from direct mapping the reference mask to the query image. The HOG features can then be computed from the predefined areas in the same manner as described in the training phase in previous section.

Only the corresponding areas identified for each HOG computed from the image, are compared with each other. Descriptors of each of the areas are computed and for a detected object $\mathbf{o}$, all masks representing particular state $\mathbf{s}$ are considered in the recognition process. This calculation is quite simple for objects of few states and simple

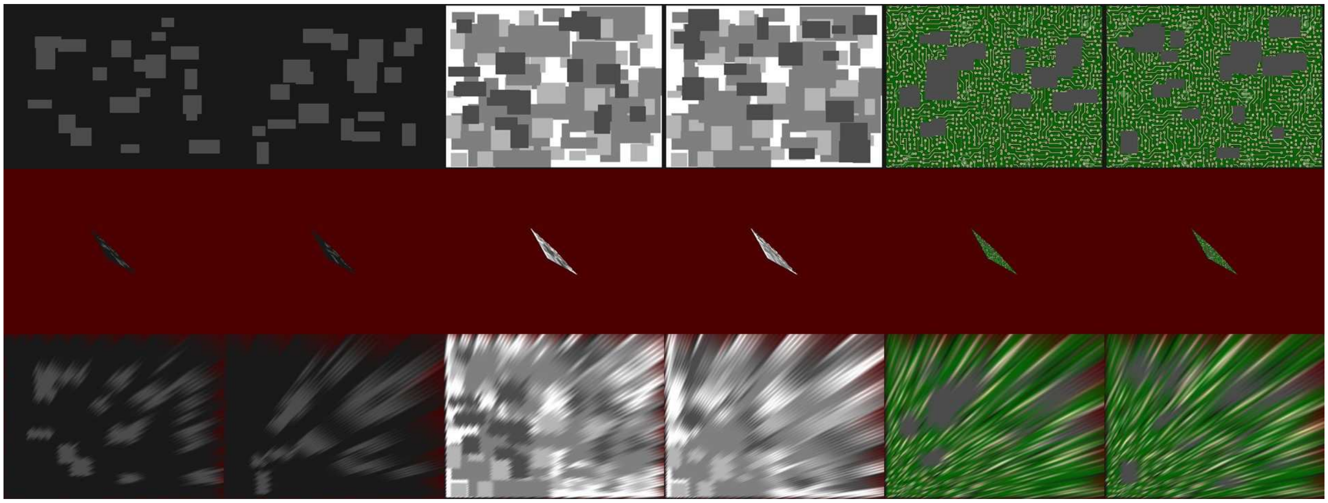

Figure 5. Example of an artificial object's region of interest (first row), its appearance after the forward rotation in *x* and *y* axis by 85°(middle row), and the output after the inverse transformation (last row). From left to right: two instances of twenty rectangles generated on a still background, twenty rectangles generated on a background with grey rectangles, and twenty rectangles generated on a background with electronic circuit picture.

masks, however may become increasingly computationally demanding for more complicated objects. The squared distance to each code-word, represented by a HOG descriptor, is then computed and the object state with the minimum distance is selected as the most suitable candidate.

3. Experimental Setup

3.1. Dataset

For the evaluation of the algorithm two types of data were used: artificially created images and real-world images. In the artificial data regions of size 300x400 pixels with three different backgrounds were created, in the real-world data, regions of 300×200 pixels were created.

The artificially created data contain three objects where the target subarea differed in its background. The first object included a solid background in the target area, the second type included a white background area with several rectangles of two shades of grey colour, and the last type included a background area comprising of a variety of electronic circuitry and components. The backgrounds were the same for each instance of the object. On top of the background, a specific number of dark rectangles were generated. The main goal of the algorithm was to identify the instance based on the structural appearance and so different colours were not considered. The side lengths of the generated rectangles were in the range of 10 to 45 pixels. The dataset consisted of 7 variations determined by the number of the rectangles; specifically in this study 3, 5, 10, 20, 30, 40 and 50 rectangles were generated. For each set of rectangles 30 variations were generated summing up to 210 sample images per each of three objects with various background: white, electronic circuit, and grey rectangles, i.e. 630 in total. The evaluation then proceeded both different numbers of rectangles and different placements of the rectangles for each of the three objects.

The real-world image dataset was created manually for a specific use-case, namely electronics rack maintenance. The objective in this use-case was to verify if a correct board has been inserted in a specified position. For this dataset 9 different positions and 8 different boards were considered. Each position was evaluated separately during the tests so that 9 objects, each with 8 + 1 instances were created for the same single rack. The additional instance represents the possibility of the specific position to be empty. Example of both datasets will be given later in the section devoted to the experiments.

3.2. Experiments

The datasets described in the previous section were used in the experiments described in the current section. This included training images of 2D objects (the targets that were to be recognized) with binary masks representing the areas where the object states differ, and test images containing one of the training objects with a given state.

A variety of image transformations were generated to simulate differing angles of view and distances, and to include the information loss that occurs with the application of such transformations. All the projections and reprojections were generated using OpenGL where the object was placed in the origin of coordinates and the distance of the camera to the object was selected in the way that the

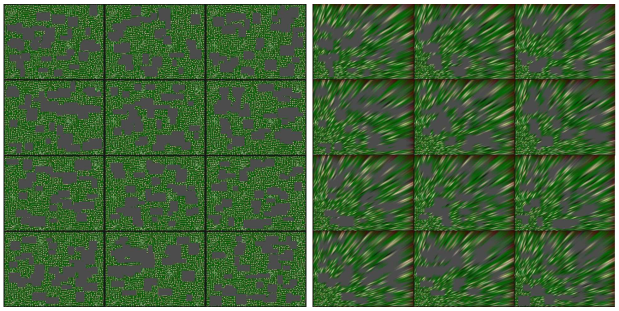

Figure 6. 12 out of 30 samples with electronic circuit background and 40 randomly generated rectangles. Left side: original images, right side: corresponding images after forward and backward rotation around *x* a *y* axes by 80°. Achieved recognition accuracy on this particular 30-sample subset: 87%.

object covers most of the scene visible by the camera with a field of view of 120°. The image was then transformed in the way simulating changes in *x* and *y* angles of view, particularly 60°, 70°, 80°, 85°, and 87°. Examples of such test objects are shown in Figure 5.

A description of the HOG parameters that were set for the experiments is given below. Only one HOG descriptor was computed per region so that the window size was equal to the size of the mask region. The region size was adapted to meet the size requirements of the HOG features derived from the block size and stride. The block shape was held constant as a square and its size varied from 8×8 to 48×48 pixels (whilst always being a multiple of eight). Cell size was always determined to be a quarter of the block size, i.e. in the range between 4×4 and 24×24 pixels. The stride was set to be equal to the cell size. Nine directions of gradient computing with keeping its orientation were used.

4. Results

The experiments tested on artificial data were evaluated separately for all the data types that were used for testing the algorithm performance. The results were further separated on the basis of the image transformation that was used during the experiments, i.e. different angles of view. Similar tests were performed on the real-world data, i.e. rack maintenance. Samples of testing data for both artificial and real-world data are depicted in Figure 6 and Figure 7, respectively. Note the information loss caused by the image warping.

The quantitative results for each number of states per object are given in Table 1. The table summarizes the results for the artificial datasets for various amounts of deformation, i.e. the angle of camera view on the object.

The results in Table 1 indicate the accuracy achieved with the proposed algorithm when applied to the artificial and real dataset with various camera angles of view. The artificial data scenarios tested included: those where the same number of rectangles had been randomly generated, i.e. the rectangles covered approximately the same percentage of the area; and those with a varying number of generated rectangles, i.e. various size of the area covered by rectangles. Three different artificial objects with a distinct background in the considered area were also directly compared. The results clearly show a significant decrease in the accuracy that was achieved with an increasing angle of view, i.e. increasing the deformation in the region of interest. However, the accuracy was still nearly 100% with an angle of view of 80° for objects with still and squared backgrounds even when the deformation was significant. For circuitry background, the accuracy was starts decreasing around 50° angle of view. At an angle of view of 85° most of the information was lost, which is apparent in the re-projection shown in Figure 6, and therefore the resulting accuracy was below 50% for circuitry background, nevertheless still above 70% for half of the experiments performed on the artificial data. Note that there was 30 different object states for the first group of experiments with same coverage of randomly generated rectangles, and 8 different object states for the varying number of generated rectangles.

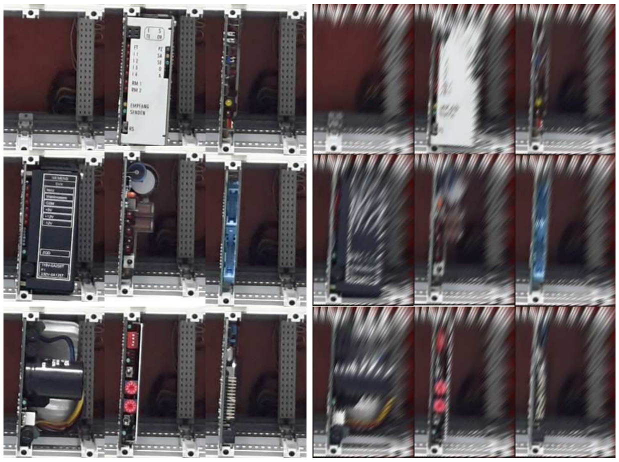

Figure 7. Samples of ROIs for a specific mask of rack maintenance data. Left side: original images, right side: corresponding images after forward and backward rotation around *x* and *y* axes by 80°. Achieved recognition accuracy on this 9-sample subset: 67%.

In the tests performed on the real-world data, particularly rack maintenance, there were 9 object states and these tests were carried out on 9 different slot positions. The accuracy of the proposed method decreased below 50% after the distortion caused by forward and backward transformation by 80° in *x* and *y* direction.

Table 1. Accuracy of correct identification of the object state for three different types of backgrounds: white, grey rectangles, and a picture of electronic circuit; for 30 object states for several camera angles of view.

Angle of view /	Same number of rectangles			Varying number of rectangles			Real-world
accuracy (%)	white	rectangles	circuit	white	rectangles	circuit	images
50°	100	100	91	100	100	100	93
60°	100	100	86	100	100	98	88
70°	100	100	73	100	100	91	72
80°	100	96	50	100	100	68	46
85°	87	76	21	34	92	40	31
87°	47	40	8	20	62	29	23

5. Discussion

From an analysis of the results presented in section 4, the proposed method was shown to be only partially resilient to image distortion that was caused by image transformation. Different viewpoint angles of the camera that are common in typical AR-usage were simulated through the applied image transformation. This approach was not suited to the simulation of changes of geometry of 3D objects as a result of changes in the viewpoint and this was not considered in the present study. Nevertheless, the impact of distortions of planar objects on the proposed algorithm was considered to be covered in the presented experiments.

The results shown in Table 1 indicate that the accuracy decreased with increasing deformation caused by increasing the angle of view, which is a somewhat intuitive conclusion. These results also suggest that the performance achieved even for large angles of view may be sufficiently accurate depending on the type of usage. When examining more closely the cases that were depicted in Figure 6, it was hard to identify the correct object state even with the human eye since almost all of the information concerning the object in the image was lost. The accuracy of the algorithm is strongly influenced by the preceding steps of the object detection and its alignment since an accurate alignment is a critical assumption within the algorithm. The proposed method did not behave well in general with two types of use-cases: firstly in the case of very similar structures; and secondly in the case where only the object colour is changed as this was not capable of being discriminated by the type of features used by the algorithm.

The benefits of the algorithm developed in the present study lie in both its simplicity and speed and also in its robustness to changes in light conditions and its high accuracy with use-cases where the structural changes were distinguishable. The drawback of the proposed approach is the necessity of some manual work during the initialization process where masks are prepared. As described in section 2.2, this limitation is already being tackled by several researchers and will likely be overcome entirely in the near-future. It is expected that future studies will enable the extension of the method to be applied to structural changes of 3D objects. State recognition of 3D object is still possible with the algorithm proposed in this paper, but, the angle of view during the recognition has to be consistent to the angle of view during the training image capturing.

6. Conclusion

In this paper an algorithm has been presented for automated object stated recognition with application in augmented reality-based industrial solutions, particularly focused on automatic verification of the maintenance and manufacturing tasks. The algorithm is based on Histogram of Oriented Gradients features. The authors have evaluated the algorithm for applications in augmented reality in the industrial environment and tested the proposed method on both artificial and real world data. The performance achieved showed its suitability to meet the requirements for the targeted application. High accuracy was achieved with large image deformations as the result of varying the angle of view, even for cases where it was hard to identify the object state by the human eye. A limitation of the current approach lies in the manual work needed during the data preparation, i.e. the manual masking of the region of interest that differentiates the object states. The automation of this process is already under development in the authors lab and part of future studies. The method presented in this study has demonstrated adequate results for the recognition of states of planar objects. Since current method can identify state of 3D objects only if the angle of view is preserved, in future investigations, the authors will explore the extension of the algorithm proposed in this paper to the recognition of objects whose states are defined by 3D structural changes for varying angle of view.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 680633.

References

[1] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. In *ECCV*, 2006.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[3] B. Girod, V. Chandrasekhar, and Y. A. R. R. Grzeszczuk. Mobile visual search: Architectures technologies and the emerging mpeg standard. *IEEE Trans. Multimedia*, 18(3):86–94, 2011.

[4] K. M. Inc. Konica minolta wearable communicator, 2017.

[5] X. Liu, J. He, and B. Lang. Multiple feature kernel hashing for large-scale visual search. *Journal of Foo*, 47(2):748–757, 2014.

[6] D.-G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, 1999.

[7] P. M. Roth and M. Winter. Survey of appearance-based methods for object recognition. Technical report, Institute for Computer Graphics and Vision, 2008.

[8] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: an efficient alternative to sift or surf. In *WACV*, 2015.

[9] C. S. S. Lazebnik and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[10] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin. A self-adjusting approach to change detection based on background word consensus. In *ICCV*, 2011.

[11] C. M. Sukanya, G. Roopa, and V. Paul. A survey on object recognition methods. *International Journal of Computer Science Engineering and Technology (IJCSET)*, pages 48–52, 2016.

[12] X. Zhang, Z. Li, L. Zhang, W. Ma, and H. Shum. Efficient indexing for large scale visual search. In *ICCV*, 2009.