TELKOMNIKA, Vol.12, No.4, December 2014, pp. 1132~1141
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v12i4.388

Review of Local Descriptor in RGB-D Object Recognition

Ema Rachmawati*[1], Iping Supriana[2], Masayu Leylia Khodra[3]
School of Electrical Engineering & Informatics, Institut Teknologi Bandung
*Corresponding author, e-mail: ema.rachmawati22@students.itb.ac.id[1], iping@stei.itb.ac.id[2],
masayu@stei.itb.ac.id[3]

Abstract

The emergence of an RGB-D (Red-Green-Blue-Depth) sensor which is capable of providing depth and RGB images gives hope to the computer vision community. Moreover, the use of local features began to increase over the last few years and has shown impressive results, especially in the field of object recognition. This article attempts to provide a survey of the recent technical achievements in this area of research. We review the use of local descriptors as the feature representation which is extracted from RGB-D images, in instances and category-level object recognition. We also highlight the involvement of depth images and how they can be combined with RGB images in constructing a local descriptor. Three different approaches are used in involving depth images into compact feature representation, that is classical approach using distribution based, kernel-trick, and feature learning. In this article, we show that the involvement of depth data successfully improves the accuracy of object recognition.

Keywords: *RGB-D images, local descriptor, object recognition, depth images*

1. Introduction

Object recognition is an important problem in computer science, which has attracted the interest of researchers in the fields of computer vision, machine learning and robotics [1]. The core of building object recognition systems is to extract meaningful representations (features) from high-dimensional observations such as images, videos and 3D point clouds [2]. Satisfactory results have been achieved by using a variety of methods, applications and standard benchmark datasets. Nevertheless, object recognition of daily objects in a scene image is still an open problem. The major challenges in a visual object recognition system are divided into two groups, which are related to system robustness and computational complexity and scalability. Belong to the first group is the challenge in handling intra-class variations in appearance (different appearance from a number of objects of the same category) and inter-class variations. Instances of the same object category can generate different images caused by a variety of variables that influence illumination, object pose, camera viewpoint, partial occlusion and background clutter. While the challenges belonging to the second group include very large objects of different categories, high-dimensional descriptors and difficulties in obtaining labelled training samples without any ambiguity etc. [3].

To address these two challenges, [3] argues that there are three aspects involved, namely modelling appearance, localization strategies and supervised classification. The focus of the researchers was trying to develop techniques and algorithms in those three aspects in order to improve the visual object recognition system performance. Among these three aspects, modelling appearance is the most important aspect [3]. Appearance modelling is focused on the selection of features that can handle various types of intra-class variations and can capture the discriminative aspects of the different categories. Furthermore, [4] also stated that "*the next step in the evolution of object recognition algorithm will require radical and bold steps forward in terms of the object representations, as well as the learning and inference algorithm used*".

The emergence of the RGB-D sensor (Microsoft Kinect, Asus Xtion, and PrimeSense), which is relatively cheap, promises to improve performance in object recognition. The sensor is capable of providing a depth image for each pixel so that the image information is abundant. RGB-D sensor has an RGB camera and an infrared camera and projector, so it can capture colour images and the depth of each pixel in the image. These two factors are very helpful for the image processing field that was always dependent on the colour channels of the image [5], [6]. By using the depth channel for foreground segmentation or complementary information on

Received September 8, 2014; Revised November 20, 2014; Accepted December 1, 2014

image intensity, there have many object recognition researches using RGB-D images with very significant results, when compared to using only the RGB camera, as can be seen in [1],[2],[7]–[10].

In general, the ways to represent image features were divided into two groups, i.e. globally and locally [11]. In the representation of local features [12], a number of features in a region that surround the target object were extracted to represent the object, so that object can be recognized in partial occlusion. Until now, we have found four other survey-like papers to introduce a local descriptor [13]–[16]. Zhang et al. [13] classified feature detectors and feature descriptors and their implementation on computer vision problems. That paper did not discuss the involvement of depth data on a feature descriptor. Paper [14] compared the performance of descriptors computed for local interest regions. The descriptor was computed on greyscale images and did not consider an object's colour. While paper [15] compared available descriptors in PCL (Point Cloud Library) [17], explaining how they work, and made a comparative evaluation on the RGB-D object dataset [7]. The major difference between this article and [13],[14],[16] is that [13],[14],[16] explain some local descriptors from RGB images and their implementation in computer vision, while this article intends to give insights into how researchers exploit RGB and depth images in constructing local descriptor in an RGB-D object recognition system, especially research that uses the RGB-D Object dataset [7]. The papers reviewed in this article were categorized into three approach according to the technique used in representing feature, that is classical technique (distribution based), kernel method, and feature learning. We show that unsupervised feature learning in constructing feature representation offers a great opportunity to be developed, in regarding to the depth image from RGB-D images, in order to capture better shape features. The rest of this article is organized accordingly. Specifically, we summarize the RGB-D Object Dataset in Section 2; describe the local descriptor in Section 3; and summarize and analyse the use of some local descriptors in RGB-D based object recognition in Sections 4 and 5. This survey concludes in Section 6.

2. RGB-D Object Dataset

RGB-D Object Dataset [7] is similar to the 3D Object Category Dataset presented by Savarese et al. [18], which contains 8 object categories, 10 objects in each category, and 24 distinct views of each object. But the RGB-D Object Dataset is on a larger scale, with RGB and depth video sequences of 300 common everyday objects from multiple view angles totalling 250,000 RGB-D images. RANSAC plane fitting [19] was used to segment objects from the video sequences.

Objects are grouped into 51 categories using WordNet relations hipernim-hiponim and are a subset of the categories in ImageNet [20]. This dataset does not only consist of textured objects such as soda cans, cereal boxes or bags of food, but also consists of textureless objects such as bowls, cups of coffee, fruit, and vegetables. Objects contained in the dataset are commonly found in homes and offices, where personal robots are expected to operate. Objects are arranged in a tree hierarchy with the number of instances of each object category found in each leaf node, ranged from 3-14 instances for each category.

3. Local Descriptor

In the representation of local features, a number of features in the region that surrounds the target object were necessary for the object to be recognized in partial occlusion [11]. This is achieved through the following steps: (1) Finding a distinctive keypoint, (2) Defining the region around the keypoint, (3) Extracting and normalizing content of region, (4) Building local descriptors of normalized region, and (5) Local descriptor matching. Zhang et al. [3] classifies the description of visual features into three groups, namely the pixel level, patch level and region level. At the pixel level, features are calculated for each pixel separately. The popular description in this group is grey-scale value that indicates the intensity of pixels along with colour vector. At patch level, a patch/support region/neighbourhood of a point is a local small sub-window that surrounds some points of interest in the image plane or scale pyramid, which can be in sparse sampling using the keypoint detector [13],[21]–[23] or in dense sampling on a regular grid. Patches, which are typically small in size, made a patch level descriptor also known as a local feature descriptor. Some popular patch level descriptors include SIFT [24], SURF [25] and the filter-bank response (Gaussian function, Gabor functions, wavelets).

Patch size, which is often too small to be able to accommodate part or all of the object means a greater region is needed to capture the more relevant visual cues. Region is a group of interconnected pixels in an image. Region can be a segment with regular or irregular shape. The region can even be the whole image. Descriptions at region level are usually developed with the purpose of capturing the most discriminating visual properties of the target category (or component of target categories) and maintain robustness in dealing with intra-class variations. Based on these objectives, the modern system of categorization adapts histogram-based representation at the region level, such as the BoF (Bag-of-Features) and HOG (Histograms of Oriented Gradients), which are usually built on contrast-based local features such as gradient, which are invariant to the lighting or colour variations. Shape cues are also often captured and described at the region level for object recognition, such as contour or edge fragments, shapelets etc. Colour features are sometimes used as a category cue, because each category has a relatively constant colour. Some descriptors at the region level are the BoF [26], HOG [27], GIST [28]–[30], and shape features [31].

4. Local Descriptor in RGB-D based Object Recognition System

A feature descriptor was built from a number of input images. In classical approach, the features were extracted from local image patches around detected interest points or using a fixed grid using a powerful method such as SIFT, SURF or Texton etc. Then, a learning algorithm, usually a technique in machine learning, was applied on those feature vectors in order to classify them into some predefined categories (see Figure 1). Those feature descriptors have been successfully used in many applications; however, they tend to be difficult to design and can not be easily adapted if there is additional information. Therefore, [32] conducted an experiment to generalize features based on orientation histogram to a broader class of so-called kernel descriptor. In constructing a kernel function one can combine knowledge that humans already have about the specific problem domain. Kernel methods can operate in a high-dimensional feature space by simply computing the inner products between the images of all pairs of data in the feature space [33].

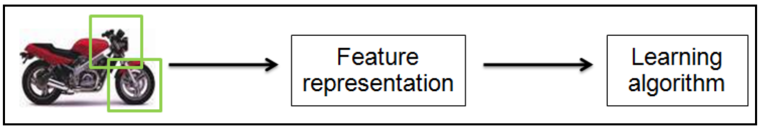

Figure 1. Common Pipeline in Feature Representation

The performance of machine learning techniques relies heavily on the selection of features representation of the application domain. So most of the effort in deploying machine learning algorithms lies in the design of pre-processing and transformation data that produces data representations that can support the effectiveness of machine learning techniques. This feature engineering process requires human intelligence and prior knowledge to overcome the weaknesses of the learning algorithm, which is unable to extract and classify discriminative information from the data. Representation learning seeks to learn representations of the data and is making it easier for the process of extracting useful information when building a classifier or other predictors [34]–[36]. Various methods to learn low-level features from raw data (feature learning) have been produced by the machine learning community, i.e. Deep Belief Network [37], deep Boltzmann machine [38], convolutional deep belief network [39] etc. Various researches that implement feature learning have also successfully demonstrated impressive accuracy. Coates et al. [40] successfully proved that good image features can be learned efficiently using standard unsupervised learning techniques (see Figure 2). However, those applications are still somewhat limited to 2D images, typically in grey-scale. [1],[9] successfully showed very good results on RGB-D object recognition using an unsupervised feature learning method in building feature representation.

To the best of my knowledge, there were six papers [1],[2],[7]–[10] that have proposed a new feature descriptor in RGB-D object recognition that use the RGB-D Object Dataset. They can be categorized into three groups based on current trends in machine learning; that is the kernel-trick approach, feature learning approach and the distribution-based approach. A summary of the feature representation approach can be seen in Table 1.

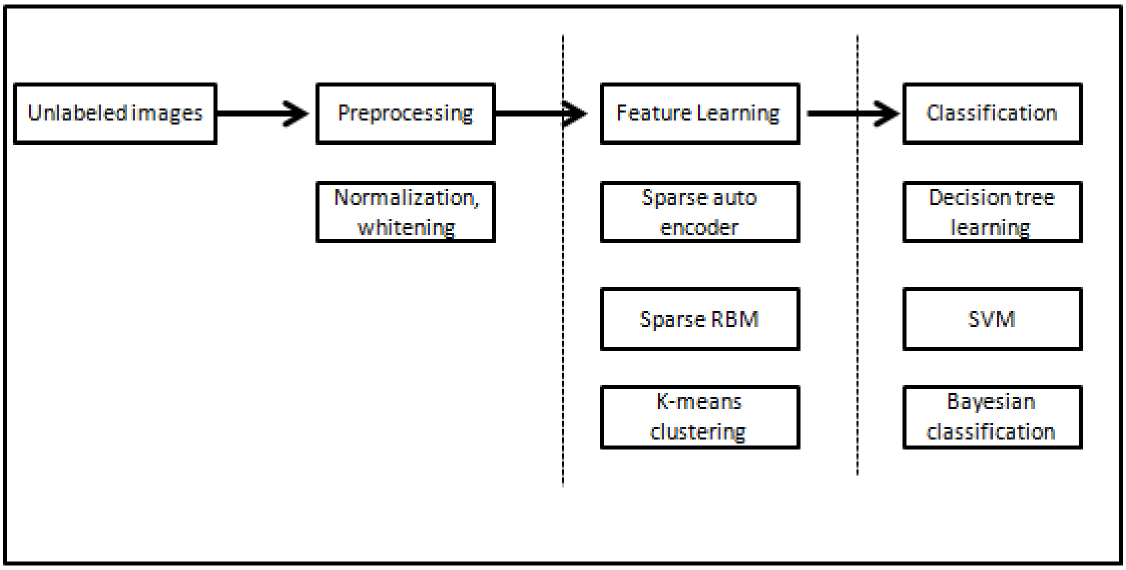

Figure 2. Feature Learning in Feature Representation

4.1. Kernel Descriptor

Lai et. al [7] build a large-scale and hierarchical multi-view dataset, namely RGBD Object Dataset, for the purposes of object recognition and detection. In addition, [7] also introduce object recognition and detection technique based on RGB-D, with combination of color and depth information. Feature extraction method commonly used in RGB image also implemented here, i.e the spin images [41] - to extract the shape feature - and SIFT [24] - to extract the visual features. Shape feature extraction is generated from the 3D location coordinates of each pixel depth. Spin images is computed from a set of 3D coordinates of a random sample. Each spin image is centered on a 3D coordinate and save the coordinates of the spatial distribution of its neighbouring points. Distributions were made in 2-dimensional histogram of size 16 x 16, which invariant to rotation. Spin images are used to compute EMK features [42] using random Fourier set. EMK (Efficient Match Kernel) features estimates gaussian kernel between local features and provide a continuous similarity value. Spatial information are combined to create grid size of 3 x 3 x 3, then 1000 EMK feature dimension is computed for each cell. One hundred principal component taken by PCA (Principal Component Analysis) on EMK features in each cell. Width, depth, and height from 3D bounding box is also added into the shape feature, so we get a 2703-dimensional shape descriptor.

The visual features are extracted from the RGB data. SIFT are extracted from 8 x 8 grid. Texton histogram feature [43] are extracted to obtain texture information, using a gaussian filter response oriented. Texton vocabulary built from a set of images on LabelMe [44]. Color histogram, mean, and standard deviation from each color channel is added as well as the visual features. The process of recognition of the object category and object instances performed using SVM (linear kernel [45] and gaussian kernel [46]) and Random Forest [47].

The experimental results showed that the overall visual features are more useful than shape features for category-level and instance-level recognition. However, shape feature is relatively more useful for category level recognition. From this research we can concluded that the combination of shape features and the visual features produce high performance in the category level recognition using any classification method. A special note was given to the alternating-contigous-frame technique, in which only uses visual features can produce high accuracy. While on leave-sequence-out technique, the combination of the visual and shape features can significantly improve the accuracy, but not as good at contiguous alternating frames.

Table 1. Summary of Feature Representation

Approach	Paper	Extracted feature from RGB Image	Extracted feature from depth Image	Descriptor Explanation	Accuracy on RGBD Object Dataset	
					Instance (%)	Category (%)
Kernel-trick	Kernel Descriptor [7]	SIFT, texton histogram, colour histogram, mean, standard deviation	Spin image (using 3D location), 3D bounding box (width, depth, height)	Using EMK to generate fixed-length feature vector and perform PCA on EMK features	Depth: 46.2 RGB: 60.7 RGB+Depth:74.8	Depth: 64.7 ± 2.2 RGB: 74.5 ± 3.1 RGB+Depth: 83.8 ± 3.5
	Depth Kernel Descriptor [8]	Colour, gradient, LBP	Edge feature: aggregation of all distance attribute pairs Size feature: Distance between each point and the reference point of the point cloud Shape feature: kernel spin and kernel PCA	Building kernel descriptor from RGB and depth images. Using pyramid EMK to integrate spatial information.	Depth: 54.3 RGB: 78.6 RGB+Depth : 84.5	Depth: 78.8 ± 2.7 RGB: 77.7 ± 1.9 RGB+Depth: 86.2 ± 2.1
	Hierarchical Kernel Descriptor [2]	Colour, gradient, LBP	Same as [8]	Define kernel descriptor over kernel descriptor. Spatial information considered by integrating center position of each patch.	Depth:46.8 RGB: 79.3 RGB+Depth: 82.4	Depth: 75.7 ± 2.6 RGB: 76.1 ± 2.2 RGB+Depth: 84.1 ± 2.2
Feature learning	Convolutional K-Means [1]	Position coordinate	Position coordinate	Interest point was detected using SURF; learning feature using Convolutional K-Means (unsupervised learning).	RGB+Depth: 90.4	RGB+Depth: 86.4 ± 2.3
	Unsupervised Feature Learning using HMP [9]	Grey-scale intensity, RGB values	Depth values, 3D surface normal	Learning feature using HMP (unsupervised learning) via K-SVD)	Depth: 51.7 RGB: 92.1 RGB+Depth: 92.8	Depth: 81.2 ± 2.3 RGB: 82.4 ± 3.1 RGB+Depth: 87.5 ± 2.9
Classic	Histogram of Oriented Normal Vectors [10]	N/A	Histogram of tangent plane orientation	Modifying : zenith & azimuth angle	N/A	RGB+Depth: 91.2 ± 2.5

4.2. Hierarchical Kernel Descriptor

Bo et. al [2] attempted to improve the accuracy of object recognition performed by [7], using kernel descriptor [32] in hierarchical way. The use of kernel descriptor [32] is very effective when used in conjunction with EMK and non-linear SVM, thus not suitable for large data. Therefore [2] tried to use kernel descriptor recursively to generate features, by changing the pixel attribute into patch level features as well as adding depth information to the descriptor. Kernel descriptors used in the RGB image to represent object features consist of gradient match kernel (based on pixel gradient attributes), colour kernel (based on pixel intensity attributes), and shape kernel (based on local binary pattern attributes). The principle of kernel descriptor adapted to the depth image by treating the depth image as greyscale images. Gradient and shape kernel descriptor can be extracted easily. While the colour kernel descriptor is extracted by previously multiplying the depth value with root s, where s is the number of pixels from object mask. Features constructed from 2-layer hierarchical kernel descriptors: (1) First layer: same as the kernel descriptor from the image patch size of 16 x 16; (2) Second layer: 1000 basis vectors are used for the Gaussian kernel.

4.3. Depth Kernel Descriptor

Bo et. al [8] conducted other techniques to improve the accuracy of image-based object recognition RGB-D, namely to create a kernel depth descriptor. Bo et. al [8] extract 5 depth kernel descriptors to represent recognition cues including size, 3D shape, and the edges of objects (depth) within a framework. The idea derived from the use of kernel descriptor on RGB images [32] in which discretizing pixels attributes was not necessary. Similarities between image patches was calculated based on kernel function, that is match kernel, that will compute average of similarity value between all pairs of pixel attributes in 2 image patches. Depth image was first converted to a 3D point cloud by mapping each pixel to the corresponding 3D coordinate vector.

The use of kernel descriptor that converts pixel attribute into patches features, making the process of generating various features from recognition cues can be done easily. Kernel descriptor for the gradient and the local binary pattern kernel [32] is extracted from the depth image. Kernel gradient and local binary pattern kernel is a representation of edge features. Gradient and local binary pattern kernel features is extracted from a 16 x 16 image / depth patch by 8 pixel spacing. Computing gradient was same as that used in the SIFT. PCA dimensions was set 50, while the other was set 200. Size descriptor, kernel PCA descriptors (shape descriptors), and spin kernel descriptor (shape descriptors) were extracted from 3D point clouds. On kernel size, for each interest point will be taken not more than 200 3D point coordinates. As for the kernel PCA and spin, distance from local region to interest point was set at 4 cm, and the number of neighbours is not more than 200 point coordinates.

Objects were modelled as a set of local kernel descriptor. Aggregating local kernel descriptors into object level features was conducted using EMK pyramid [42], [48]. Kernel local descriptor is mapped in a low dimensional features space and will further build on object-level features by taking the average value of the resulting features vector. Object recognition accuracy increased significantly by implementing the five descriptors. In addition, [8] also successfully demonstrated that the performance of kernel features exceeded the performance of 3D spin images features. From the results of experiments conducted, [8] found that the depth features is worse than RGB features at instances level recognition. This is because different instances in the same category can have a shape that is almost similar. So the combination of depth features by RGB features can improve the recognition accuracy. While on recognition the category, the performance of depth kernel descriptor is quite comparable to the image kernel descriptor, which indicates that the depth information is as important as the visual information for category recognition.

4.4. Convolutional K-Means Descriptor

Blum et. al [1] improve the accuracy of object recognition based on RGBD image, by proposing an algorithm that is able to automatically recognize image features, in which colour and depth is encoded in a compact representation. Blum et. al [1] introduce a new descriptor, namely convolutional k-means descriptor, which automatically learn the response of a number of neighbouring features of interest point which was detected. Phases in the process of formation of the K-Means Convolutional Descriptor can be described as follows:

1. Learning feature responses
   a. *Unsupervised learning* [40], learning a set of feature responses of a number of input vectors.
   b. Normalization of all patches by subtracting with the average value and dividing by the standard deviation. PCA whitening transformation [49] is then performed on the image patches.
   c. Image patches clustering using k-means.
2. Interest point detection, using SURF [25] to extract SURF corner.
3. Descriptor extraction.

4.5. Hierarchical Matching Pursuit for Depth Data

Bo et. al [9] tried to improve the accuracy of object recognition algorithm by adapting HMP (Hierarchical Matching Pursuit) in two layer [50]. HMP is adapted for RGB-D images, by learning dictionaries and encodes features using all RGB-D data (greyscale, RGB, depth, and channel surface normal). HMP uses sparse coding to perform learning (unsupervised) hierarchical features representation from the RGB-D data. HMP will build dictionaries from patch and depth image using the K-SVD [51] to represent objects as a sparse combination of codeword. Furthermore, hierarchical features was built using orthogonal matching pursuit and spatial pyramid pooling. So that HMP can be used for RGB-D images, here are the steps should be done:

1. Learning features on the colour and depth images based on the concept of sparse coding. Sparse coding will perform dictionaries learning, that is the representation of data with the linear combination (and sparse) of data entry on dictionaries. Data entry was pixel values of image patches size of 16 x 16.
2. HMP build hierarchical features of dictionaries from the result of (1) by applying the orthogonal matching pursuit encoder recursively and performing spatial pyramid max pooling performed on sparse code on each layer of the hierarchical HMP.

At the instance level recognition, features obtained from the learning on colour image successfully improve the performance compared to features obtained from a grey-scale image. In addition, features of the first layer is better (fine-grained). In contrast to the category-level recognition, in which the features of the second layer better (coarse-grained). Based on the experimental results, learning dictionaries separately for each colour channel produces better accuracy than perform learning together.

4.6. Histogram of Oriented Normal Vectors

HONV (histogram of oriented normal vectors) was designed by [10] to capture the characteristics of the 3-D geometry from image depth. Without relying on texture, the object is expected to be recognized by taking into account this 3D surface. To reduce noise in depth image at the time of pre-processing, Gaussian filter is used. HONV is histogram-based features, such as HOG features [27] and LBP. Object surface is assumed to represent object categories information, because the object surface can be described by a tangent plane orientation (i.e normal vector on each coordinate surface). Characteristics of 3D geometry can be represented as a local distribution from orientation of the normal vector. Tang et. al [10] made decline in the formula, which shows that the normal vector can be represented as an ordered pair of azimuth and zenith angles, which can be easily calculated from the gradient of depth image. HONV is the concatenation from the local histogram of azimuth and zenith angles, so it can be used as features in the object detection/classification.

Normal vector on the position p = (x, y) is the cross product of two vectors tangent on tangent plane. Through a decline in the formula, [10] get the formula of normal vector on pixel (x, y, d (x, y)). Spherical coordinates are used to encode orientation information with the representation of zenith and azimuth angles. Phases of getting HONV features are as follows: (i) Dividing detection window in the size of m x n cells. The orientation of the normal vector on each cell is computed and made into histograms. Feature vectors (i x j dimensional) will be formed from each cell, i as a representation of the zenith angle, and j as a representation of azimuth angle, with I = J = 8; (ii) Final feature was obtained by combining HONV features of each cell.

5. Analysis

Image is a data source with special characteristics. Each pixel in the image represents a measurement. Beside it having high dimensions, the vector representation of images typically indicates strong correlation between a pixel and its neighbours. Kernel methods have proved successful in many areas in computer vision, mainly because of their interpretability and flexibility [33]. In the kernel method, a feature descriptor is constructed by comparing pixel orientations or colour intensities in a kernel representation. In the kernel representation, we simply compute the inner products between all pairs of data in the feature space [52]. Kernels are typically designed to capture one aspect of the data, i.e. texture, colour or edge etc. So, [2,7,8] design and use a new kernel method in order to capture all aspects of the image to describe an object. Their experiment results showed that their kernel approach has proven to be successful in representing features from RGB and depth images. There is significant improvement in the accuracy of instance and category level recognition using their kernel trick, as can be seen in Table 1.

Hand-designed features such as SIFT and HOG only capture low-level edge information. Although it has proven difficult to design features that capture mid-level cues (e.g. edge intersections) or high-level representation (e.g. object parts) effectively, they support many successful object recognition approaches. In addition, the recent developments in deep learning have shown how hierarchies of features can be learned in an unsupervised way directly from data. The use of deep learning has proved successful in lowering state-of-the-art error rate on the ImageNet object recognition 1000-class benchmark [53]. In this paper we show that unsupervised feature learning has a high potential in building a feature descriptor that is more discriminative than the kernel method [1,9]. Unlike the kernel method, which often uses a nonlinear classifier in the classification process, the feature descriptor generated through feature learning typically can be easily learned using a linear classifier [1,9]. This approach enables learning meaningful features from RGB as well as depth data automatically. It can be seen from the experimental results (Table 1) that the accuracy of instance object recognition successfully achieves enough margin compared to the use of kernel-trick [2,7,8]. The accuracy of category object recognition is also increased, although not as much as the increase in instance object recognition. These results are extremely encouraging, indicating that current recognition systems can be significantly improved without having to design features carefully and manually. This work opens up many possibilities for learning rich, expressive features from raw RGB-D data.

Unlike the five other papers reviewed in this paper, Tang et al. [10] did not recognize an object at instance-level in their experiment, but focused on exploiting depth images to capture shape features for object category recognition. The local surface of an object was captured relating to the histogram of azimuth angle and zenith angle to describe its 3D shape. This idea achieved the state-of-the-art object category recognition on the RGB-D dataset as can be seen in Table 1.

6. Conclusion

We have presented a survey highlighting the current technical achievement of a local descriptor on object recognition based on RGB-D images, as well as its influence on the accuracy of object recognition. Exploration of local descriptors on depth image combined with the RGB image, which was conducted by some research goes into this article. From various studies it appears that the presence of the depth image has a positive effect on object recognition. Extraction of local features on depth images can be used to help recognize objects. In addition, the combination of the visual features and shape features of RGB image and depth in a certain descriptor has proved to be capable of producing an object recognition system with very high accuracy on the RGB-D Object dataset. From the three approaches described in this article, accuracy of instance recognition involving depth features using feature learning approach shows more significant improvement than using kernel method. Whereas the classical approach using normal vector distribution achieves highest accuracy in category recognition.

References

[1] Blum M, Springenberg JT, Wulfing J, Riedmiller M. *A learned feature descriptor for object recognition in RGB-D data*. IEEE International Conference on Robotics and Automation (ICRA). 2012: 1298–1303.

[2] Bo L, Lai K, Ren X, Fox D. *Object recognition with hierarchical kernel descriptors*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2011: 1729–1736.

[3] Zhang X, Yang Y-H, Han Z, Wang H, Gao C. Object Class Detection: A Survey. *ACM Comput Surv.* 2013; 46(1): 10:1–10:53.

[4] Andreopoulos A, Tsotsos JK. 50 Years of object recognition: Directions forward. *Computer Vision and Image Understanding.* 2013; 117(8): 827–891.

[5] Cruz L, Lucio D, Velho L. *Kinect and RGBD Images: Challenges and Applications*. SIBGRAPI Conference on Graphics, Patterns and Images Tutorials. 2012: 36–49 .

[6] Liu H, Philipose M, Sun M-T. Automatic objects segmentation with RGB-D cameras. *Journal of Visual Communication and Image Representation.* 2014; 25(4): 709–718.

[7] Lai K, Bo L, Ren X, Fox D. *A large-scale hierarchical multi-view RGB-D object dataset.* IEEE International Conference on Robotics and Automation (ICRA). 2011: 1817–1824.

[8] Bo L, Ren X, Fox D. *Depth kernel descriptors for object recognition*. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2011: 821–826.

[9] Bo L, Ren X, Fox D. *Unsupervised Feature Learning for RGB-D Based Object Recognition*. In International Symposium on Experimental Robotics (ISER). 2012.

[10] Tang S, Wang X, Lv X, Han T, Keller J, He Z, et al. *Histogram of Oriented Normal Vectors for Object Recognition with a Depth Sensor.* In: Lee K, Matsushita Y, Rehg J, Hu Z, editors. Computer Vision – ACCV 2012, vol. 7725, Springer Berlin Heidelberg. 2013: 525–38.

[11] Grauman K, Leibe B. Visual Object Recognition. Synthesis Lectures on Artificial Intelligence and Machine Learning. 2011; 5: 1–181.

[12] Zhang J, Marszałek M, Lazebnik S, Schmid C. Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. *International Journal of Computer Vision.* 2006; 73(2): 213–238.

[13] Li J, Allinson NM. A comprehensive review of current local features for computer vision. *Neurocomputing.* 2008; 71(10-12): 1771–1787.

[14] Mikolajczyk K, Schmid C. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 2005; 27(10): 1615–1630.

[15] Mikolajczyk K, Schmid C. Scale & Affine Invariant Interest Point Detectors. *International Journal of Computer Vision.* 2004; 60(1): 63–86.

[16] Tian D ping. A Review on Image Feature Extraction and Representation Techniques. *International Journal of Multimedia and Ubiquitous Engineering.* 2013; 8(4): 385–396.

[17] Aldoma A, Marton Z-C, Tombari F, Wohlkinger W, Potthast C, Zeisl B, et al. Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation. *IEEE Robotics Automation Magazine*. 2012; 19(3): 80–91.

[18] Savarese S, Fei-Fei L. *3D generic object categorization, localization and pose estimation*. IEEE 11th International Conference on Computer Vision (ICCV). 2007: 1–8.

[19] Fischler MA, Bolles RC. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM.* 1981; 24(6): 381–395.

[20] Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L. *ImageNet: A large-scale hierarchical image database*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2009: 248–255.

[21] Tuytelaars T, Mikolajczyk K. Local Invariant Feature Detectors: A Survey. *Found Trends Comput Graph Vis.* 2008; 3(3): 177–280.

[22] Miksik O, Mikolajczyk K. *Evaluation of local detectors and descriptors for fast feature matching*. 21st International Conference on Pattern Recognition (ICPR). 2012: 2681–2684.

[23] Schmid C, Mohr R, Bauckhage C. Evaluation of interest point detectors. *International Journal of Computer Vision.* 2000; 37(2): 151–172.

[24] Lowe DG. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision.* 2004; 60(2): 91–110.

[25] Bay H, Ess A, Tuytelaars T, Van Gool L. Speeded-Up Robust Features (SURF). *Comput Vis Image Underst.* 2008; 110(3): 346–359.

[26] Hauptmann AG. Representations of Keypoint-Based Semantic Concept Detection: A Comprehensive Study. *IEEE Transactions on Multimedia.* 2010; 12(1): 42–53.

[27] Dalal N, Triggs B. *Histograms of oriented gradients for human detection*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2005; 1: 886–893.

[28] Torralba A. Contextual Priming for Object Detection. *International Journal of Computer Vision.* 2003; 53(2): 169–191.

[29] Oliva A, Torralba A. The role of context in object recognition. *Trends in Cognitive Sciences.* 2007; 11(12): 520–527.

[30] Naji D, Fakir F, Bencharef O, Bouikhalene B, Razouk A. Indexing Of Three Dimensions Objects Using GIST Zernike & PCA Descriptors. *IAES International Journal of Artificial Intelligence* (IJ-AI). 2013; 2(1):1–6.
[31] Belongie S, Malik J, Puzicha J. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2002; 24(4): 509–522.
[32] Bo L, Ren X, Fox D. Kernel Descriptors for Visual Recognition. *Advances in Neural Information Processing Systems.* 2010.
[33] Lampert CH. Kernel Methods in Computer Vision. *Found Trends Comput Graph Vis.* 2009; 4(3): 193–285.
[34] Erhan D, Bengio Y, Courville A. Why does unsupervised pre-training help deep learning? *The Journal of Machine Learning Research.* 2010; 11: 625–660.
[35] Bengio Y. Learning Deep Architectures for AI. *Found Trends Mach Learn.* 2009; 2(1): 1–127.
[36] Bengio Y, Courville A, Vincent P. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 2013; 35(8): 1798–1828.
[37] Hinton GE, Osindero S, Teh Y-W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput* . 2006; 18(7): 1527–1554.
[38] Salakhutdinov R, Hinton G. *Deep Boltzmann Machines*. Proceedings of the International Conference on Artificial Intelligence and Statistics. 2009; 5: 448–455.
[39] Lee H, Grosse R, Ranganath R, Ng AY. *Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations.* Proceedings of the 26th Annual International Conference on Machine Learning (ICML) . 2009: 1–8.
[40] Coates A, Lee H, Ng AY. *An analysis of single-layer networks in unsupervised feature learning*. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. 2011: 215–223.
[41] Johnson AE, Hebert M. Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes. *IEEE Trans Pattern Anal Mach Intell* .1999; 21(5): 433–449.
[42] Bo L, Sminchisescu C. Efficient Match Kernel between Sets of Features for Visual Recognition. In: Bengio Y, Schuurmans D, Lafferty JD, Williams CKI, Culotta A, editors. *Advances in Neural Information Processing Systems* 22, Curran Associates, Inc. 2009: 135–143.
[43] Leung T, Malik J. Representing and Recognizing the Visual Appearance of Materials Using Three-dimensional Textons. *Int J Comput Vision.* 2001; 43(1): 29–44.
[44] Russell BC, Torralba A, Murphy KP, Freeman WT. LabelMe: A Database and Web-Based Tool for Image Annotation. *Int J Comput Vision*. 2008; 77(1-3): 157–173.
[45] Chang C-C, Lin C-J. LIBSVM: A Library for Support Vector Machines. *ACM Trans Intell Syst Technol*. 2011; 2(3): 27:1–27:27.
[46] Fan R-E, Chang K-W, Hsieh C-J, Wang X-R, Lin C-J. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research.* 2008; 9: 1871–1874.
[47] Breiman L. Random Forests. *Mach Learn.* 2001; 45(1): 5–32.
[48] Grauman K. The Pyramid Match Kernel : Efficient Learning with Sets of Features. *Journal of Machine Learning Research.* 2007; 8: 725–760.
[49] Hyvärinen A, Oja E. Independent component analysis: algorithms and applications. *Neural Networks*. 2000; 13(4-5): 411–430.
[50] Bo L, Ren X, Fox D. Hierarchical Matching Pursuit for Image Classification: Architecture and Fast Algorithms. In: Shawe-Taylor J, Zemel RS, Bartlett PL, Pereira F, Weinberger KQ, editors. *Advances in Neural Information Processing Systems* 24, Curran Associates, Inc. 2011: 2115–2123.
[51] Aharon M, Elad M, Bruckstein A. K -SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Transactions on Signal Processing.* 2006; 54(11): 4311–4322.
[52] Wahyuningrum R, Damayanti F. Efficient Kernel-based 2DPCA for Smile Stages Recognition. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2012; 10(1): 113–118.
[53] Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks. In: Bartlett P, Pereira F c. n., Burges C j. c., Bottou L, Weinberger K q., editors. *Advances in Neural Information Processing Systems* 25, 2012: 1106–1114.