Proceeding of the IEEE
International Conference on Robotics and Biomimetics (ROBIO)
Shenzhen, China, December 2013

A Vision-based Robotic Grasping System Using Deep Learning for 3D Object Recognition and Pose Estimation

Jincheng Yu, Kaijian Weng, Guoyuan Liang and Guanghan Xie

Abstract— **Pose estimation of object is one of the key problems for the automatic-grasping task of robotics. In this paper, we present a new vision-based robotic grasping system, which can not only recognize different objects but also estimate their poses by using a deep learning model, finally grasp them and move to a predefined destination. The deep learning model demonstrates strong power in learning hierarchical features which greatly facilitates the recognition mission. We apply the Max-pooling Convolutional Neural Network (MPCNN), one of the most popular deep learning models, in this system, and assign different poses of objects as different classes in MPCNN. Besides, a new object detection method is also presented to overcome the disadvantage of the deep learning model. We have built a database comprised of 5 objects with different poses and illuminations for experimental performance evaluation. The experimental results demonstrate that our system can achieve high accuracy on object recognition as well as pose estimation. And the vision-based robotic system can grasp objects successfully regardless of different poses and illuminations.**

I. INTRODUCTION

It is researchers' dream to create an intelligent robot which can identify objects, grasp them and move them to the target locations just like human. That's of great importance in the work of household service, industrial production, or space exploration. Robotic grasping, however, is extremely difficult because of unknown objects and poses. Alvaro Collet and Manuel Martinez achieved robust object recognition and pose estimation with MOPED, a framework for Multiple Object Pose Estimation and Detection [1]. But this model-based object recognition method has difficulty in the model building stage. In [2], a feature-based framework that combines spatial feature clustering, guided sampling for pose generation, and model up-dating for 3D object recognition and pose estimation is used. However, like some other feature-based methods, it relies on feature discriminability for correct matches and on the robust estimator capabilities for outlier rejection [3]. Nowadays, more and more approaches are presented for robotic grasping. Ellen Klingbeil proposed a novel algorithm for grasping unknown objects given raw depth data obtained from a single frame of a 3D sensor [4]. Although the approach requires no explicit models of objects and does not need a training phase, it may fail to identify and grasp the objects which are too large or too small to provide dependable depth measurements. Kai Huebner and Danica Kragic proposed an algorithm that efficiently wraps given 3D data points of an object into primitive box shapes by a fit-and-split algorithm based on Minimum Volume Bounding Boxes [5]. Some others built hierarchical representations based on edge and texture information, which were sparse but powerful descriptions of the scene, and they got edge-based and surface-based grasps based on this representation [6]. Unfortunately, edge and texture based methods are usually susceptible to noises.

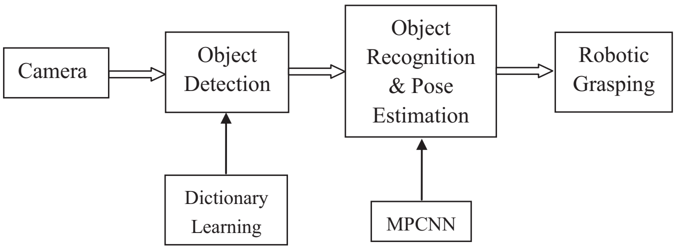

Fig. 1. Data flow chart of our system.

In this paper, we propose a new system which can not only recognize objects but also estimate their poses by using deep neural network model. Recently, deep learning model shows great potential in machine vision and data mining. Although it becomes more and more popular for object recognition [7] [8], it's rarely reported in object pose estimation, specifically for robotics. In our system, the Max-pooling Convolutional Neural Network (MPCNN) is applied in the learning model to deal with both tasks all at once. Deep learning model, however, does not work well for object detection. Therefore, we propose a new object detection method to segment objects from background before recognition. A database comprised of 5 objects (cup, coke bottle, pen, box and screwdriver) with various poses is built for training and testing in our experiment. A detailed description on the database will be presented in section IV.

The data flow chart is shown in Fig. 1. The objects are firstly detected by a new dictionary-based learning method. Then we use MPCNN to recognize objects and estimate their poses. At last, the robot controller moves the arm, grasp the object and move them to a predefined destination. In this system, we follow Zhang's method [9] for camera calibration.

Our paper is organized as follows: section II will present the algorithm of the object detection; object recognition and pose estimation with MPCNN will be given in section III;

Jincheng Yu, Kaijian Weng and Guoyuan Liang are with Guangdong Provincial Key Laboratory of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. Shenzhen. China; Jincheng Yu is also with Guangdong University of Technology, Guangzhou, China. jc.yu@siat.ac.cn, kj.weng@siat.ac.cn and gy.liang@siat.ac.cn

Guanghan Xie is with Guangdong University of Technology, Guangzhou, China. xiegh64@126.com

978-1-4799-2744-9/13/$31.00 ©2013 IEEE

section IV will focus on our experiment while conclusions and future works will follow in section V.

II. Object Detection

Visual object detection is the most important step for robotic grasping. Many methods have been reported so far. As almost all the object recognition methods rely heavily on the accuracy of foreground object detection, efficient and reliable methods are imperative. This section is concentrated on presenting a new segmentation method based on dictionary learning to detect the objects from the background.

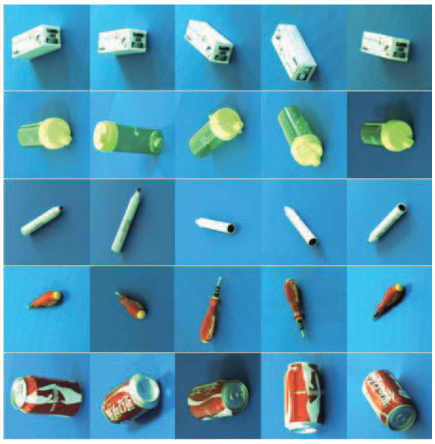

Fig. 2. Objects with different poses.

Our dictionary is built by using K-means clustering for the color component of the images with background and objects. Clustering is a way to separate groups of components. It treats each component as having a location in feature space, and finds partitions such that components within each cluster are as close to each other as possible and as far from components in other clusters as possible [10]. In the robotic grasping system, we pick 5 objects including cup, box, coke bottle, screwdriver and pen, illustrated in Fig. 2. Moreover, 30 color images under different illuminations are captured for each object. Each image is normalized to size 100×100, and transformed to a vector with size 10000 and dimension 3. Therefor, there are $n(n = 10000 \times 180)$ pixels in our initial cluster. We use K-means with the Euclidean distance metric to cluster the initial cluster into m clusters. Then, with this dictionary, we can get the key features of objects by comparing the statistics of the projections of background and foreground. It is easy to segment the objects from background by key features. We empirically set $m = 30$.

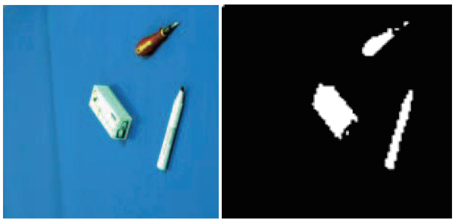

Fig. 3. Segmentation result (left: the original image captured by camera with size 1624×1324, right: the segmentation result with the projection vectors).

Algorithm 1 The algorithm for extracting the key features.

```
Definition: The cluster center is denoted by $C_{ki}$, $X_{si}$ refers
    to the input pixel. The distances between foreground
    object and each center is denoted by $D_o(s,k)$. The distances
    between background and each center is denoted
    by $D_b(s,k)$, where $k(k = 1, 2, ..., m)$ is the number of
    the cluster center, $i(i = 1, 2, 3)$ is the input dimension.
    The dimension is 3 because the input pixel contains 3
    color channels. $s(s = 1, 2, ..., n)$ is the number of the
    input.
1: Calculate the distance between each sample and each
    center. The distance is:
        $D(s,k) = \sqrt{\sum_{i=1}^{3} (X_{si} - C_{ki})^2}$
2: If $D_o(s,k) < T$, then
    $o\_center_k = o\_center_k + 1$
    If $D_b(s,k) < T$, then
    $b\_center_k = b\_center_k + 1$
    Where $T$ is the threshold. Here, we set $T = 30$,
    $o\_center_k$ and $b\_center_k$ represent the statistic of fore-
    ground and background respectively.
3: If $\frac{b\_center_k}{o\_center_k} < \varepsilon$ then
    $C_{ki}$ is the key feature of the foreground, where $\varepsilon$ is an
    infinitesimal.
```

The segmentation result with the key features is shown in Fig. 3.

III. Object Recognition and Pose Estimation with MPCNN

A. Brief Introduction of MPCNN

MPCNN is a special type of Convolutional Neural Networks (CNNs) with max-pooling layers. The max-pooling layers are the amelioration of the pooling layers, similar to the complex cells in visual cortex [11], whose purpose is to achieve spatial invariance by reducing the resolution of the feature maps [12]. Meanwhile, max-pooling reduces the computational complexity for upper layers by selecting superior invariant features. In the traditional model of pattern recognition, hand-crafted feature extractors are usually used by researchers to eliminate the extraneous variables and extract the relevant information from the input. However CNNs are designed to recognize visual patterns directly from pixel images with minimal preprocessing. Like almost every other neural networks CNNs are trained with a version of the back-propagation algorithm [13].

B. Object Recognition and Pose Estimation

Comparing with the traditional object recognition based on the deep learning model, we focus on the object pose estimation including object recognition. Deep learning methods have the capability of recognizing or predicting large set of

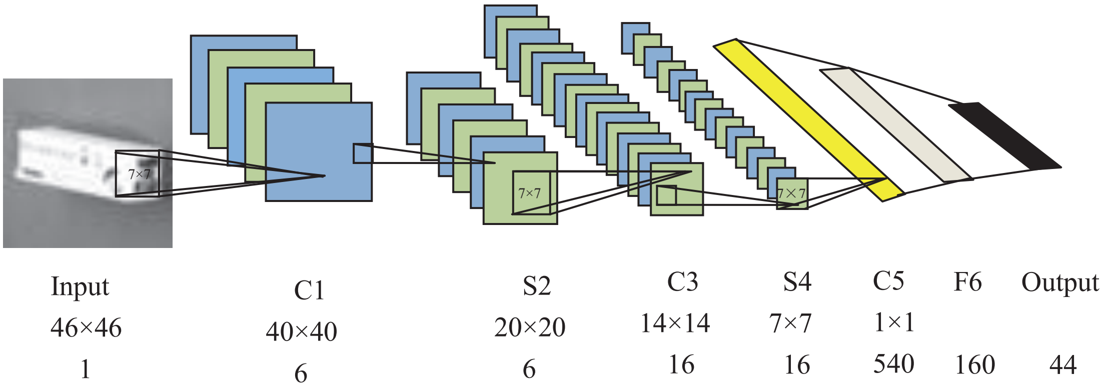

Fig. 4. MPCNN architecture in the experiment.

patterns by learning sparse features of small set of patterns. With this advantage, we can use a small set of poses to train the deep learning model, and then predict a large set of poses with the model. In our system, we assign different poses of objects as different classes. Hence, a class represents a pose of object in the MPCNN. For example, the poses of box are divided into 6 pose classes every 30°, and numbered class 1 to class 6. These 6 classes correspond to the first 6 neurons of the output layer in the MPCNN. So, we can not only recognize the object but also the pose by outputs of neurons in the last layer.

The architecture of our MPCNN indicated in Fig. 4 which is the optimal for our database.

The input layer is a raw image of size 46×46. Next is a convolutional layer C1 with 7×7 filters and 6 maps of size 40×40.The following 2×2 max-pooling layer S2 reduces the size to 20×20, convolutional layer C3 convolved with 7×7 filters has 16 maps of size 14×14. Max-pooling layer S4 reduces the size to 7×7 by max-pooling from 2×2 sliding windows. The convolutional layer C5 employs 7×7 filters and has 540 maps dimensions of 1×1 which are fully connected to 160 hidden neurons in layer F6. The output layer fully connects to the layer F6 and encodes the object class with 44 neurons.

We use all images of the training set, where all classes are evenly ranked, for training. In order to speed up the learning process, each input image is normalized to have a mean of zero and a variance of one in the input layer. In this system gradient descent is used for learning in MPCNN.

IV. EXPERIMENT

The main purpose of this paper is to verify the feasibility and effectiveness of the deep learning model for object recognition and pose estimation, and implement a new complete vision-based robotic control system for object grasping. We build a database comprised of 5 objects with various poses for experiment. As shown in Fig .3, all the 5 objects are used for pose estimation. To evaluate the precision of the robotic grasping, we select pen, screwdriver and box whose sizes are graspable for the gripper. The system set up is shown in Fig. 5. Our system employs a Basler industrial camera and a Schunk Dextrous Lightweight Arm LWA 4D mechanical arm. The camera is mounted on a movable bracket and is positioned at 35°above the horizontal.

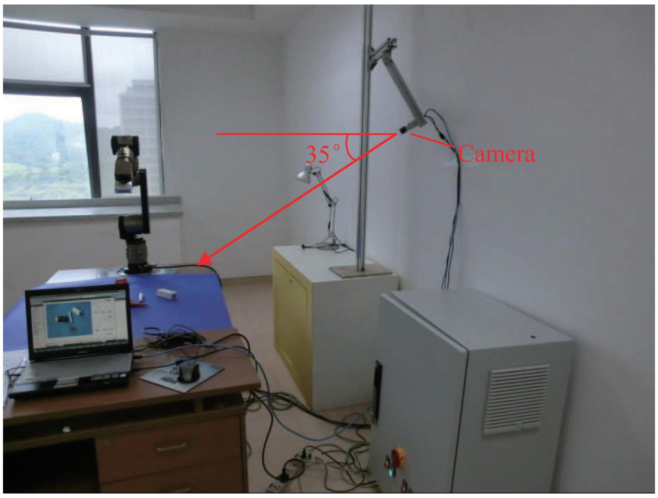

Fig. 5. The illustration of our system.

A. Database

Our database has 5 objects, includes 1600 images for training set, 1980 images for test set I and 924 images for test set II with the same background but different orientations under different illuminations. The way how we create the database by sampling in the continuous orientation space is shown in Fig. 6, where each region indicates a pose class. For coke bottle and cup, there is an extra upright pose class except for the 2D poses in the desktop plane. We should point out that our method actually has the capability to learn any 3D poses. All the samples in each pose class are sampled every 5°under 5 different illuminations in the training set while the images in test set I captured every 2°. Test set II is the subset of the test set I where the 4 poses far away from the central pose of each class are eliminated. For example, class 2 shown in Fig. 6 contains the poses sampled in the range from 15°to 45°. The central pose is 30°. The 4 poses eliminated are 16°, 18°, 42°and 44°.

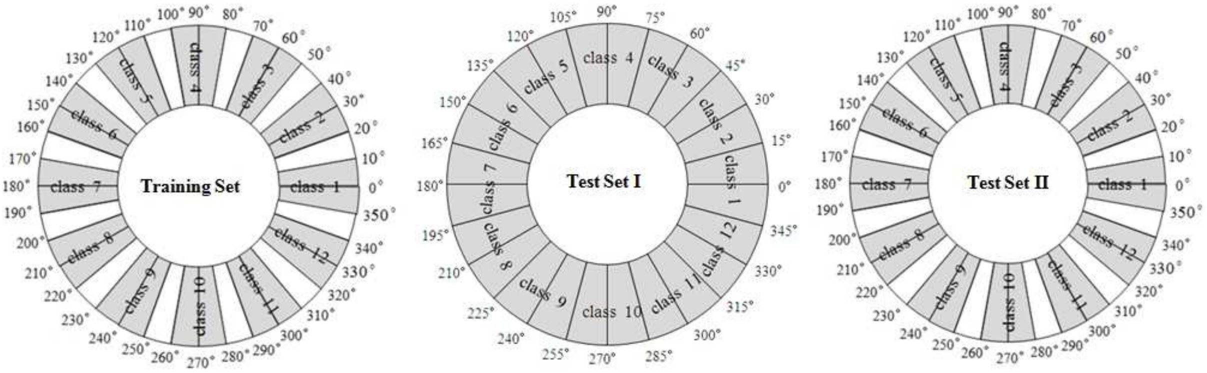

Fig. 6. Sampling diagram (the samples in each class of training set are sampled every 5°while the samples in each class of test set I and test set II are sampled every 2°).

TABLE I
EVALUATION OF OBJECT RECOGNITION AND POSE ESTIMATION

Object	Class Number	Test I		Test II	
		Total Number	Error Rate	Total Number	Error Rate
Box	6	270	0.133	126	0.079
Screwdriver	12	540	0.048	252	0
Pen	6	270	0.019	126	0
Cup	13	585	0.055	273	0
Coke Bottle	7	315	0.035	147	0
Total	44	1980	0.055	924	0.011

TABLE II
EVALUATION OF ROBOTIC GRASPING

Object	Total Trials	Recognition		Grasp	
		Failures	Failures Rate	Failures	Failures Rate
Box	60	4	0.067	3	0.050
Screwdriver	60	4	0.067	2	0.033
Pen	60	0	0	0	0

In these pose classes, all images are transformed into gray images with size 46×46. The poses for symmetrical objects such as box and pen are grouped in the same class. Finally our database contains 44 pose classes.

B. Object Recognition and Pose Estimation Result

The purpose of creating two test sets is to show the different performances for pose set with different sizes. Table I shows that the recognition rate of the full test set I reaches 94.5% while the recognition rate of the full test set II reaches 98.9%. The result of box has the highest error rate, the reason may be that the patterns of box in different poses have higher similarity than the patterns of other objects. Results of test I and test II indicate that errors most likely happened when the pose is far away from the central pose of training set. Since there are 30°between adjacent class centers, hopefully the recognition rate will increase if the test poses are closer to the central pose. Fig. 7 shows the estimation results for several objects in different poses.

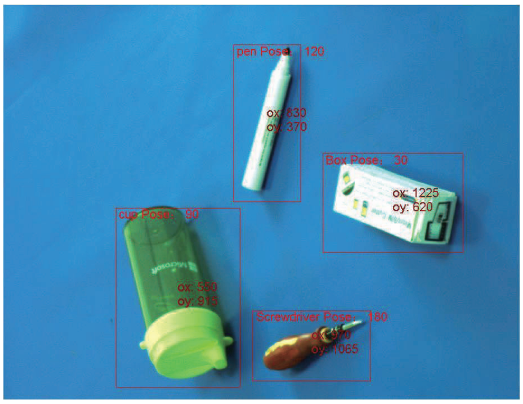

Fig. 7. Object recognition and pose estimation of different objects (pen, box, cup and screwdriver).

C. Robotic Grasping

This is the final stage of the experiment where the system changes the position and orientation of the gripper in order to grasp the objects. Our robot has a gripper installed at the end of its arm, as shown in Fig. 9, the "grasp-depth"

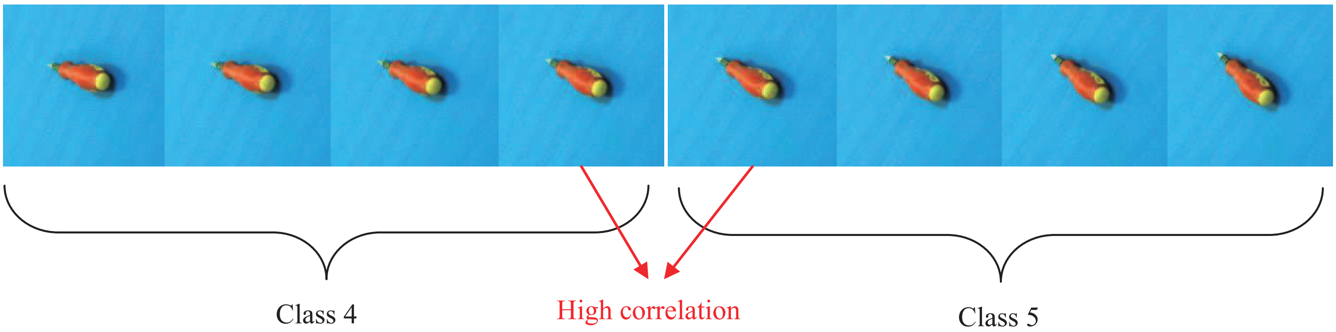

Fig. 8. Images of two adjoining pose classes (The adjacent poses from different class have a high correlation).

$H = 60mm$ and grasp "spread" $L = 76mm$. We notice that in order to have a reliable grasp, the size of the object should be a little bit smaller than the maximum spread of the gripper. Therefore, we select pen, box and screwdriver for the grasping experiment because the coke bottle and cup are too large for the gripper.

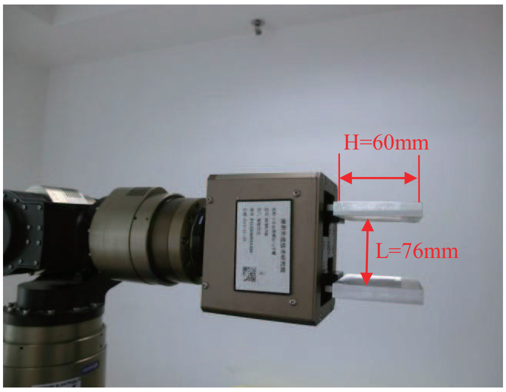

Fig. 9. "Grasp-depth" and grasp "spread".

The path planning for the arm has been achieved by a classical method [14]. Here, we focus on how to adjust the gripper pose to a desired orientation for a successful grasping. Particularly, we define the center of each pose class as the target pose for the gripper, because the pose estimation tends to be more accurate if it is closer to the central pose for each class. We also designed two experimental schemas for robotic grasping. For the first one, each object is placed every approximate 10°clockwise to make 36 trials. In the second schema, grasping experiments are performed on 24 scenes consist of three objects which are at random poses. So there are 60 trials for each object, table II shows the results.

As there is a high correlation between the adjacent poses (see Fig. 8), false pose estimation tends to occur in between. However, very little effect is observed in most cases.

V. CONCLUSION AND FUTURE WORKS

In this paper, we present a vision-based robotic grasping system that is capable of object detection, recognition and grasping with different poses by using a deep learning model. We built an image database of 5 objects with various poses both for training and testing purpose. A deep learning model is then applied for object recognition and pose estimation. Experimental results demonstrate that our system can make accurate object recognition, pose estimation, as well as successful grasping. The deep learning model is proven to be effective in pose estimation from a single image of object without any 3D models or 3D information. This kind of vision-based system shows great potential in industrial and household services.

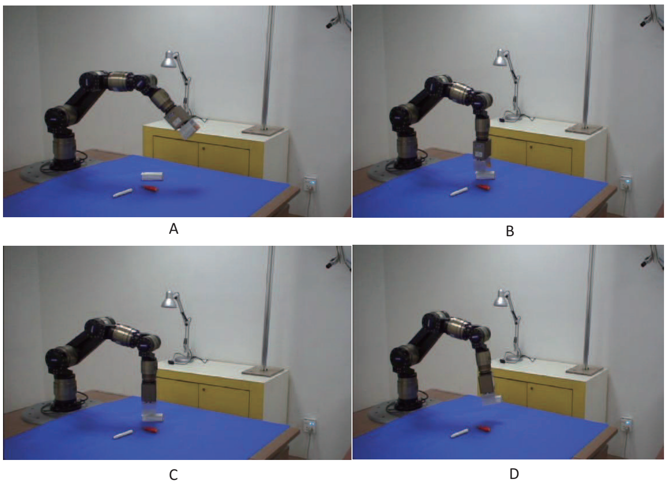

Fig. 10. Snapshots of grasping experiment (A: object detection and pose estimation; B: move to the position above the object; C: adjust the pose of the gripper; D: grasp object and move to the predefined destination.

Though the pose estimation results are expected to be more accurate by making denser classer of pose in MPCNN, there are still giant space for improving the learning model to make it more robust for images with complex background in the future.

VI. ACKNOWLEDGMENTS

This work is partially supported by Shenzhen Nanshan Research and Design Special Fund No. KC2012JSYB0050A, Guangdong Innovative Research Team Program (201001D0104648280).

References

[1] A. Collet, M. Martinez, and S. S. Srinivasa, “The MOPED framework: Object Recognition and Pose Estimation for Manipulation”, *in International Journal of Robotics Research*, 2011.

[2] Fenzi M, Dragon R, Leal-Taix L, “3D Object Recognition and Pose Estimation for Multiple Objects using Multi-Prioritized RANSAC and Model Updating”, *in Pattern Recognition*, 2012, pp. 123-133.

[3] Bhat, S., Berger, M.O., Sur, F., “Visual Words for 3D Reconstruction and Pose Computation”, *in 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, 2011, pp. 326-333.

[4] E. Klingbeil, D. Drao, B. Carpenter, V. Ganapathi, O. Khatib, and A. Y. Ng, “Grasping with application to an autonomous checkout robot”, *in Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011.

[5] K. Hbner, D. Kragic, “Selection of robot pre-grasps using box-based shape approximation”,*in IEEE Int. Conference on Intelligent Robots and Systems* , 2008, pp. 1765-1770.

[6] Mila Popovic, Gert Kootstra, Jimmy A. Jrgensen, Danica Kragic, Norbert Krger, “Grasping unknown objects using an early cognitive vision system for general scene understanding”, *in Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 987-994.

[7] Ciresan, D.C., Meier, U., Schmidhuber, J., “Multi-column deep neural networks for image classification”, *in Computer Vision and Pattern Recognition*, 2012, pp. 3642-3649.

[8] LeCun, Y., Huang, F.J., Bottou, L., “Learning methods for generic object recognition with invariance to pose and lighting”, *in Computer Vision and Pattern Recognition*, 2004, pp. 97-104.

[9] Z. Zhang, “A flexible new technique for camera calibration”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, pp.1330-1334.

[10] H. Zha, C. Ding, M. Gu, X. He, and H. Simon, “Spectral relaxation for k-means clustering”, *in Advances in Neural Information Processing Systems*, 2002.

[11] D.H. Hubel and T.N. Wiesel, “Receptive fields and functional architecture of monkey striate cortex”,*in Journal of Physiology*, 1968, pp. 215-243.

[12] D. Scherer, A. Muller, and S. Behnke, “Evaluation of pooling operations in convolutional architectures for object recognition”, *in Proc. of the Intl. Conf. on Artificial Neural Networks*, 2010, pp. 92-101.

[13] Kevin Jarrett, Koray Kavukcuoglu, Marc’Aurelio Ranzato and Yann LeCun, “What is the Best Multi-Stage Architecture for Object Recognition?”, *in Proc. International Conference on Computer Vision (ICCV)*, 2009.

[14] Gou, Zhijian, Ying Sun, and Haiying Yu, “Inverse kinematics equation of 6-DOF robot based on geometry projection and simulation”, *in Computer, Mechatronics, Control and Electronic Engineering (CMCE)*, 2010, pp. 125-128.