A Low-Cost Stereo System for 3D Object Recognition

Fabio Oleari*†, Dario Lodi Rizzini*, Stefano Caselli*
*RIMLab - Robotics and Intelligent Machines Laboratory
Dipartimento di Ingegneria dell'Informazione, University of Parma, Italy
†Elettric80 S.p.a. - Via G. Marconi, 23 42030 Viano (RE), Italy
E-mail {oleari,dlr,caselli}@ce.unipr.it

***Abstract*—In this paper, we present a low-cost stereo vision system designed for object recognition with FPFH point feature descriptors. Image acquisition is performed using a pair of consumer market UVC cameras costing less than 80 Euros, lacking synchronization signal and without customizable optics. Nonetheless, the acquired point clouds are sufficiently accurate to perform object recognition using FPFH features. The recognition algorithm compares the point cluster extracted from the current image pair with the models contained in a dataset. Experiments show that the recognition rate is above 80% even when the object is partially occluded.**

I. Introduction

The diffusion of relatively accurate 3D sensors has popularized scene interpretation and point cloud processing. Motion planning, human-robot interaction, manipulation and grasping [1] have taken advantage from these advancements in perception. In particular, identification of objects in a scene is a fundamental task when the robot operates in environments with human artifacts.

The complexity of object recognition depends on the accuracy of sensors, on the availability of shape or color information, on specific prior knowledge of the object dataset, and on the setup or the operating context. Although 3D perception is not mandatory for object recognition, the availability of the object shape can improve the recognition and allows the assessment of the object pose for further operations like manipulation. Low-cost 3D sensors broaden the application domains of shape processing and support the development of effective algorithms. Depth cameras and RGB-D sensors rely either on active stereo or on time-of-flight [2] and often provide an off-the-shelf solution for end-users that does not require complex calibration operations. However, active stereo devices like MS Kinect [3] or Dinast Cyclope [4] are sensitive to environment lighting conditions, since the perception of patterns in infrared or other domains may be noisy. A cheap stereo vision systems can be constructed using a pair of low-cost cameras. Such cost-effective solution requires to manually build the setup, to calibrate the complete system and to carefully tune the parameters to achieve a sufficiently dense point cloud. Moreover, a stereo system can be designed according to the requirements of a specific application (e.g. by adapting the baseline). Since such 3D sensor is not an active sensor, it can be used in outdoor environments.

A common requirement for a recognition algorithm is the identification of a region of interest (ROI) corresponding to a candidate object. This operation can be simplified by exploiting the specific knowledge about the setup, e.g. all the candidate objects lie on a table. Object recognition is commonly achieved by extracting features that represent a signature for a point neighborhood. Several 3D features to be extracted from point clouds or other representations have been proposed during the years. Spherical harmonic invariants [5] are computed on parametrized surfaces as values invariant to translation and rotation of such surfaces. Spin images [6] are obtained by projecting and binning the object surface vertices on the frame defined by an oriented point on the surface. Curvature map method [7] computes a signature based on curvature in the neighborhood of each vertex. Scale Invariant Feature Transform (SIFT) [8], which extracts points and a signature vector of descriptors characterizing the neighborhood, has established a standard model for several point feature descriptors and has popularized the feature constellation method to recognize objects. According to such approach the signature of an object consists of a collection of features extracted from the observation. Object recognition between the current observation and an object model is performed by matching each descriptor extracted from the observation with its closest descriptor in the model. If many pairs of similar points have consistent relative position, the comparison outcome is positive. The feature constellation method exploits both *feature similarity*, which is measured by a metric in descriptor space, and *feature proximity*. More recently, point feature descriptors designed according to the point descriptor paradigm like Normal Aligned Radial Feature (NARF) [9], Point Feature Histogram (PFH) and Fast Point Feature Histogram (FPFH) [10] have been proposed for 3D points. FPFH are computed as histograms of the angle between the normal of a point and the normals of the points in its neighborhood. Several features have been proposed and implemented in *Point Cloud Library* (PCL) [11]. These methods usually provide a parameter vector that describes the local shape. Such descriptors allow object recognition of known objects by matching a model and the observed point cloud.

In this paper, we present a low-cost stereo vision system designed for object recognition with FPFH point feature descriptors. We show its effectiveness even in presence of occlusions. This work demonstrates that this fundamental task can be performed using state-of-art algorithms on 3D sensor data acquired with generic consumer hardware costing less than 80 Euros, taking a different approach from RGB-D cameras. The stereo system has been built by mounting two Logitech C270 UVC (USB Video Class) cameras on a rigid bar. The main limitations of such sensors lie in the lack of hardware

synchronization trigger signals and of customizable optics. A flaw in frame synchronization may affect the accuracy of the disparity map. However, the overall image quality and resolution and the approximate software synchronization allow the computation of a sufficiently dense and accurate disparity image to perform object recognition. The calibration (intrinsic and extrinsic) and the computation the disparity image have been performed using the packages and libraries provided by ROS (Robotic Operating System) framework. Since scene segmentation is not the aim of this work, the system works under the assumption that all the objects to be recognized lie on a planar surface and inside a given bounded region. The object recognition algorithm is based on comparison of FPFH feature collections. In particular, the FPFH points extracted from the current point cloud are compared with the FPFH point models contained in an object dataset. The dataset consists of 8 objects observed from about 6 viewpoints. Tests have been performed to evaluate the recognition performance of the stereo system. The recognition algorithm has shown good performance with true positive rate above 80%. The effects of occlusion on recognition rate have been assessed by showing that the recognition performance is only slightly affected when the occluded part of the object is less than the 40% of its visible surface.

The paper is organized as follows. Section II illustrates the low-cost stereo system. Section III presents the algorithm for the identification of the region of interest where the object lies. Section IV presents the object recognition algorithms. Section V presents the experiments performed to assess performance and section VI discusses the results.

II. Hardware Setup

The stereo camera developed in this work (Figure 1) has been designed to be as general purpose as possible so that object recognition tasks can be performed in different scenarios. The stereo system exploits a pair of Logitech C270 webcams which offer a relatively good flexibility and quality compared to other low-cost consumer cameras. Logitech C270 webcams provide a high definition image sensor with a maximum available resolution of 1280x720 pixels. At the maximum resolution, the camera can grab frames with a frequency of 10 Hz.
This image sensor is fully compliant to UVC standard and allows the setting of some image parameters like brightness, contrast, white balance, exposure, gain and so on. Moreover, each webcam exposes a unique serial number that can be used to deploy a dedicated UDEV rule to set devices. In this way the system univocally distinguishes between left and right cameras.

The case has been built in aluminium to ensure a good strength to the stereo camera. Thus, the sensor can be mounted in mobile robots, on manipulators, and in other scenarios where it could be mechanically stressed due to vibrations or collisions.
The internal structure of the enclosure is realized in 5 mm thick aluminium and all parts are mounted with M2.0 precision screws. The webcam PCBs are fixed to the internal structure with M1.6 precision screws that use the existing holes in the boards. Moreover, the use of grover washers between nuts guarantees a robust fastening, without movements of the sensors that could compromise camera calibration.
On the bottom and on the back of the enclosure there are two 1/4“ UNC nuts fully compatible with photographic supports. The overall final dimensions are 205x44x40 mm and the total cost, webcams included, does not exceed 80 euro.

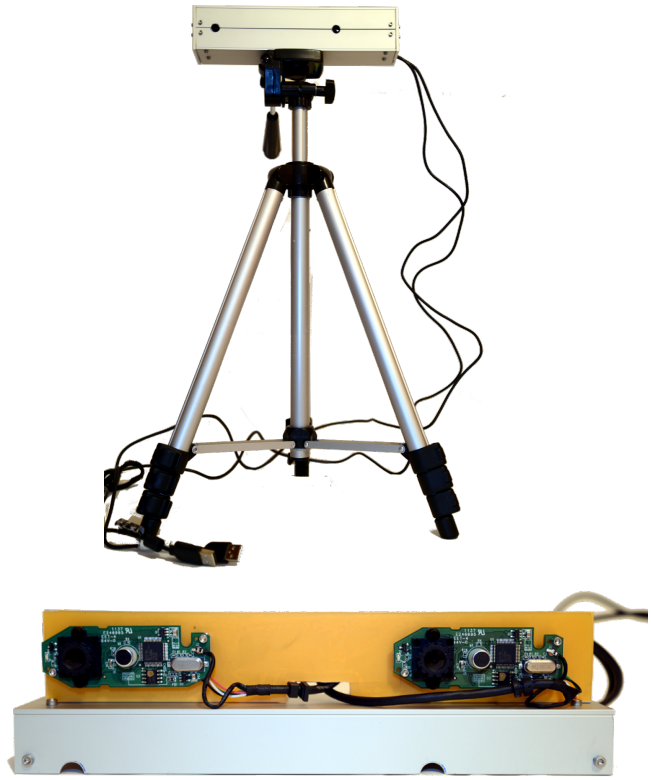


Fig. 1. Stereo camera realized with two Logitech C270 webcams.

When using non-professional devices for stereo vision, the key problem is the impossibility to synchronize the cameras with an external trigger signal. The timing incoherence of left and right frames may generate gross errors when there is a relative movement between scene and camera (moving scene and/or moving camera). To reduce this issue the webcams have been driven at the maximum frame rate available for the selected resolution which does not saturate the USB 2.0 bandwidth. In this way the inter-frame time is reduced to the minimum according to constraints imposed by the overall system architecture.

The Logitech C270 webcams can provide images in SVGA resolution (800x600) at 15 frames per second without fully occupying the USB 2.0 bandwidth.
The frame grabbing task is assigned to the ROS package *uvc_camera* and in particular to a slightly modified version of the *stereo_node* which performs the acquisition of a couple of roughly software-synchronized frames from two different devices.
Figure 2 shows the time in milliseconds needed by the driver to read both the left and right frames. The mean value is equal to 66.01 ms which corresponds to a frequency of 15Hz and the standard deviation is 3.73 ms. In the plotted sequence of 5000 samples, only 11 acquisitions were found to be not well synchronized because grabbing both the frames took

Parameter	Value
prefilter_size	9
prefilter_cap	31
correlation_window_size	15
min_disparity	0
disparity_range	128
uniqueness_ratio	15
texture_threshold	9
speckle_size	90
speckle_range	4

TABLE I. PARAMETERS OF THE STEREO RECONSTRUCTION ALGORITHM.

approximately twice the mean time. In the end, only 184 frames (corresponding to 3.68 %) were grabbed in a time higher than mean $+1\sigma$.

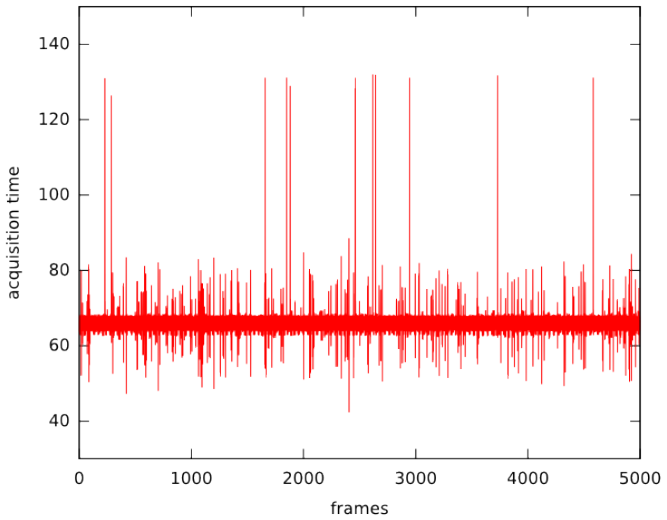
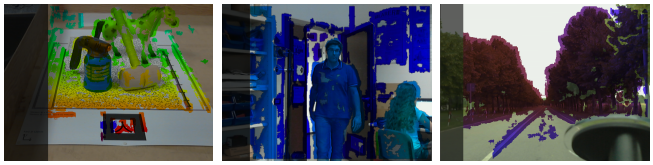


Fig. 2. Acquisition time in milliseconds for both left and right frames

III. OBJECT CLUSTER EXTRACTION

The processing pipeline starts from the acquisition of left and right frames. Then the standard ROS package *stereo_image_proc* performs disparity and computes the resulting point cloud. Parameters of the stereo reconstruction algorithm are shown in Table I and example results in different scenarios are displayed in Fig.3.



(a) (b) (c)

Fig. 3. Example results of stereo reconstruction in different scenarios: (a) working environment; (b) interior and (c) exterior/automotive.

The 3D representation of the scene is then segmented and filtered to only preserve information on the region of interest. In this work we did not focus on detection, so the extraction from the overall scene point cloud relies on the assumption that the ROI is fixed. The working environment consists of a planar surface with a known bounded region and an ARToolKit [12] marker that identifies the global *world* frame. The first step of object extraction is the geometric transformation of the point cloud, required to express the points with respect to the center of the *world*. The rototranslation matrices are obtained from the values of position and orientation of the ARToolKit marker. The resulting point cloud is then segmented with a bounding box that discards all points of table and background using *Point Cloud Library* (PCL) [13]. In the end, a *statistical outlier removal* filter is applied to discard the remaining isolated points. An example of a cluster of points resulting from the extraction process is shown in Fig.4.

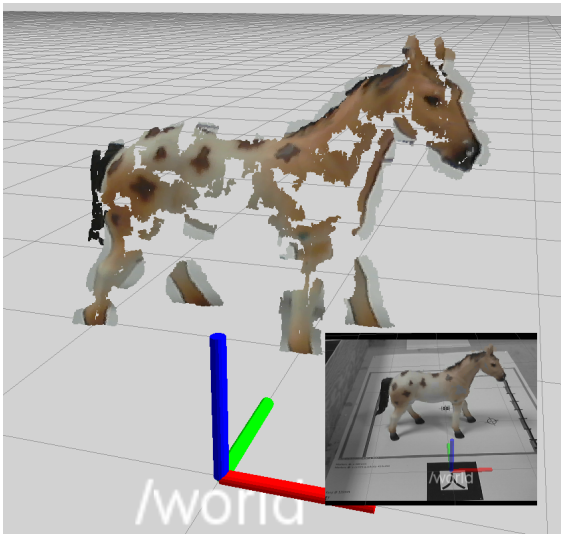


Fig. 4. Final result of the scene segmentation and filtering.

IV. RECOGNITION

Cluster recognition is the last step in the described pipeline and aims at matching a selected cluster with an entry in a dataset of known models. The dataset consists of a variable number of views for each object, taken from pseudo-random points of view, as shown in Figure 5. Each model is obtained by accumulating points from multiple frames in order to fill gaps of the cloud produced by stereo vision. Then a voxel grid filter is applied to achieve a uniformly-sampled point cloud. The recognition algorithm is based on point clouds alignment. The two clouds of the *i-th* model $\mathcal{P}_i^{mod}$ and the current object $\mathcal{P}^{obj}$ in 3D space need to be registered or aligned in order to be compared. The registration procedure computes the rigid geometric transformation that should be applied to $\mathcal{P}_i^{mod}$ to align it to $\mathcal{P}^{obj}$. Registration is performed in three different steps:

- *Remove dependency on external reference frame.* $\mathcal{P}_i^{mod}$ and $\mathcal{P}^{obj}$ are initially expressed in the reference of the respective centroids.
- *Perform initial alignment.* The algorithm estimates an initial and sub-optimal alignment between point clouds. This step is performed with the assistance of a RANSAC method that

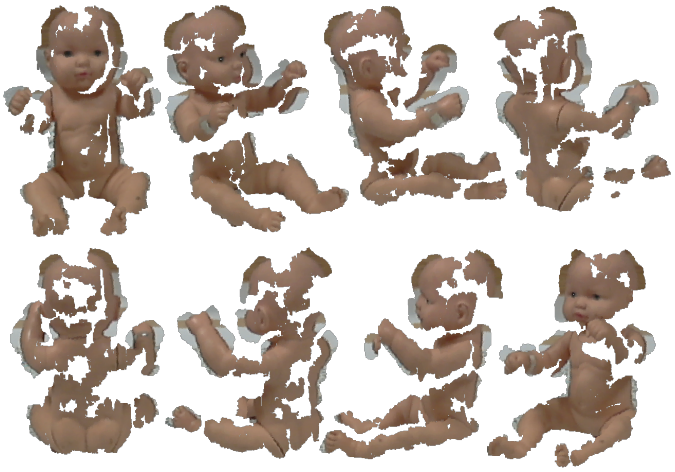


Fig. 5. Multiple models obtained from different PoV for an example object.

Algorithm 1: Registration procedure

Data:
$\mathcal{P}_i^{mod}$: Point cloud of i-th model;
$\mathcal{P}^{obj}$: Point cloud of the object to be recognized;
$\mathcal{R}$: set of search radii in FPFH features computation;
Result:
$\mathcal{P}_{i,aligned}^{mod}$: Aligned point cloud of the model;

$\mathcal{P}_c^{obj} \leftarrow$ shiftToCentroid($\mathcal{P}^{obj}$);
$\mathcal{P}_{i,c}^{mod} \leftarrow$ shiftToCentroid($\mathcal{P}_i^{mod}$);
$\mathcal{P}_{i,sac}^{mod} \leftarrow \emptyset$;
foreach $r \in \mathcal{R}$ **do**
  $\mathcal{F}_o \leftarrow$ computeFPFH($\mathcal{P}_c^{obj}$,r);
  $\mathcal{F}_m \leftarrow$ computeFPFH($\mathcal{P}_{i,c}^{mod}$,r);
  $\mathcal{P}_{i,sac}^{mod,r} \leftarrow$ getRANSACAlignment($\mathcal{P}_c^{obj}, \mathcal{F}_o, \mathcal{P}_{i,c}^{mod}, \mathcal{F}_m$);
  if getFitness($\mathcal{P}_{i,sac}^{mod,r}$) > getFitness($\mathcal{P}_{i,sac}^{mod}$) **then**
    $\mathcal{P}_{i,sac}^{mod} \leftarrow \mathcal{P}_{i,sac}^{mod,r}$;
  end
end
$\mathcal{P}_{i,aligned}^{mod} \leftarrow$ getICPAlignment($\mathcal{P}_{i,sac}^{mod}, \mathcal{P}_{obj}$);

uses FPFH descriptors as parameters for the consensus function. The computation of FPFH is performed using different search radii.

- *Refine the alignment*.
  The initial alignment is then refined with an ICP algorithm that minimizes the mean square distance between points.

The procedure is detailed in Algorithm 1 and an example result is shown in Figure 6.

Recognition is then performed by computing a *fitness* value that evaluates the overall quality of the alignment between $\mathcal{P}_{i,aligned}^{mod}$ and $\mathcal{P}^{obj}$. For each point of $\mathcal{P}^{obj}$, the algorithm calculates the square mean distance from the nearest point of $\mathcal{P}_{i,aligned}^{mod}$ and retrieves the percentage of points whose distance is below a fixed threshold δ_{th}:

$$Q = \left\{ p_i \in \mathcal{P}^{obj} : \|p_j - p_i\|^2 \leq \delta_{th}, p_j \in \mathcal{P}_{i,aligned}^{mod} \right\}$$
$$fitness(\mathcal{P}^{obj}, \mathcal{P}_{i,aligned}^{mod}) = \frac{|Q|}{|\mathcal{P}^{obj}|} \quad (1)$$

Algorithm 2: Overall recognition procedure

Data:
$\mathcal{P}^{mod}[\cdot]$: List of point cloud models;
$\mathcal{P}^{obj}$: Point cloud of the object to be recognized;
Result:
$name$: Name of the recognized object;

$\mathcal{F}_{max} \leftarrow 0$;
foreach $\mathcal{P}_i^{mod} \in \mathcal{P}^{mod}[\cdot]$ **do**
  $\mathcal{P}_{i,aligned}^{mod} \leftarrow$ performRegistration($\mathcal{P}_i^{mod}$, $\mathcal{P}^{obj}$);
  $\mathcal{F}_i \leftarrow$ getFitness($\mathcal{P}^{obj}$, $\mathcal{P}_{i,aligned}^{mod}$);
  if $\mathcal{F}_i > \mathcal{F}_{max}$ **then**
    $\mathcal{F}_{max} \leftarrow \mathcal{F}_i$;
    $name \leftarrow$ name of $\mathcal{P}_i^{mod}$;
  end
end

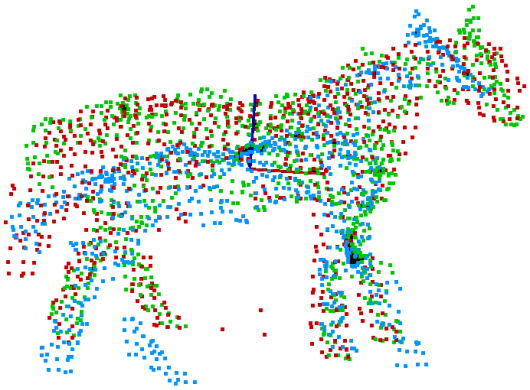


Fig. 6. Alignment of a model after RANSAC (blue) and after ICP (green) to the object (red).

Maximum fitness, equal to 100%, is obtained when all points of $\mathcal{P}^{obj}$ have a neighbour in $\mathcal{P}_{i,aligned}^{mod}$ within δ_{th}.
The algorithm is iterated for each model in the dataset and returns the recognized model with the higher fitness, as shown in Algorithm 2.

V. Results

This section presents the experiments performed to assess the performance of the recognition algorithm illustrated in the previous section. These results show the performance afforded by a low-cost stereo system in 3D object recognition. The object extraction and the recognition modules have been implemented as separated components using ROS framework. The experimental setup consists of the stereo vision system described in this paper with one of the candidate objects placed in front of the sensor. The experiments are designed to assess the object recognition algorithm, in particular when only a partial point cloud of the object is available due to noisy segmentation and occlusions. Test set consists of a fixed sequence of 2241 object point clouds taken from random viewpoints. The dataset consists of 61 models representing the 8 objects in Figure 7 (8 views for each object on average). Table II shows the confusion matrix obtained without imposing

	horse_starlet	horse	baby	big_detergent	fire	woolite	chocolate	hammer
horse_starlet	**144**	2	0	0	0	0	0	0
horse	0	**111**	2	0	0	0	0	0
baby	1	0	**128**	2	0	0	0	0
big_detergent	0	1	1	**60**	0	0	0	14
fire	5	3	3	0	**111**	1	0	0
woolite	2	0	6	8	0	**116**	0	0
chocolate	3	4	9	3	0	23	**67**	0
hammer	14	2	3	10	0	6	2	**132**

TABLE II. CONFUSION MATRIX FOR EACH CATEGORY WITHOUT OCCLUSION.

Fig. 7. Set of objects for the recognition experiment.

	# dataset	radius [mm]
test01	61	3,5,10,20,30,50
test02	32	3,5,10,20,30,50
test03	61	3,10,30
test04	24	3,5,10,20,30,50
test05	61	5,15
test06	32	5,15
test07	32	3,10,30

TABLE III. DIFFERENT TUNING OF ALGORITHM PARAMETERS.

a threshold on fitness. The classification results show that, even without a threshold on fitness to detect true negatives, correct matches largely prevail. The articulated objects (*horse*, *horse_starlet*, *baby* and *fire*) are better recognized.

The next test series takes into account parameters of the algorithm like the number of model views in the dataset and the search radius used to compute the FPFH (trials are called *test01*, *test02*, etc. in Table III). The true positive and false positive rates for the different trials are shown in Figure 8. Experimental results show the importance of including dataset models taken from multiple viewpoints. Keeping fixed all the parameters while decreasing the dataset size, the percentage of true positives decreases (see *test01*, *test02* and *test04*). On the other hand, the recognition rate is only slightly affected by restricting the set of search radii used to compute FPFH features when the full dataset of model views is available (compare *test01*, *test03* and *test05* in Figure 8). The Receiver Operating Charateristic curves in Figure 9 depict the performance of the classifier as its discrimination threshold is varied. To summarize, the webcam-based stereo system together with the recognition algorithm has shown good performance, with true positive rate above 80% provided that sufficient viewpoint models are available.

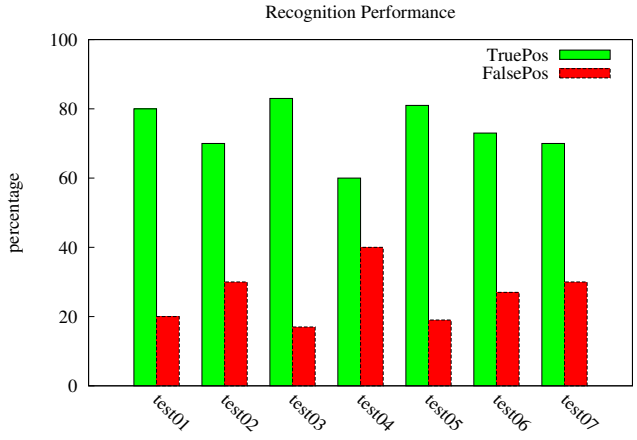


Fig. 8. True and false positive rates for the tests in Table III.

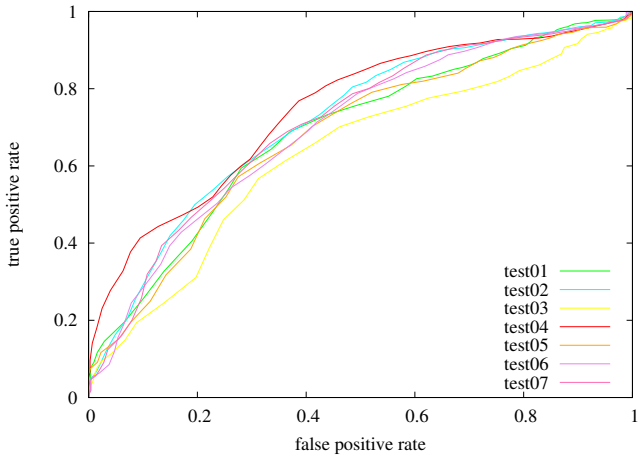


Fig. 9. ROC curves for tests test01 to test07.

We have then evaluated the recognition algorithm with partial and occluded objects. In order to have comparable results, occlusions have been artificially generated with a random procedure. The *occlusion generator* processes the original test set and for each view chooses a random point in the cloud and removes all points within a random radius. In this way it generates a new *synthetically occluded* test set with occlusions measured as percentage of removed points. Six different tests have been performed with increasing occlusions

	occl_10to20	occl_20to30	occl_30to40
Occlusions [%]	10-20	20-30	30-40
True Pos. [%]	76	70	60
False Pos. [%]	24	30	40

TABLE IV. EXPERIMENTAL RESULTS FOR TEST SET WITH OCCLUSIONS AND RECOGNITION PARAMETERS AS TEST05 [TABLE III]

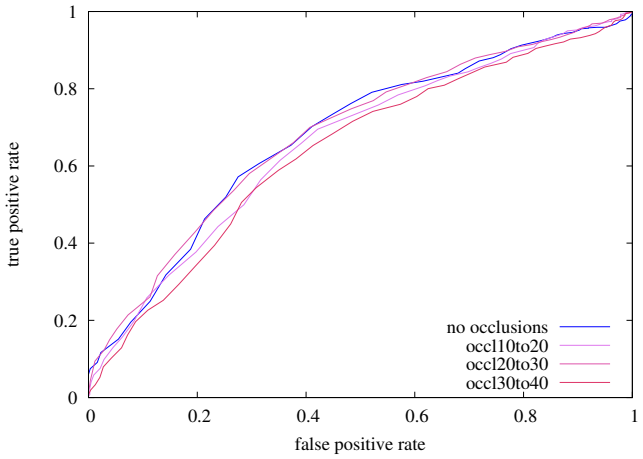


Fig. 10. ROC curves for tests with occlusions.

from 10% to 70% of the object surface perceived from the current viewpoint. Recognition results are shown in Table IV. Recognition algorithm still exhibits good performance with occlusions up to 30% with true positive rates above 70%. Performance rapidly decreases with occlusions up to 40% and then collapses with increasing percentage of occluded points. Figure 10 shows Precision-Recall curves for all tests with occlusions and a reference test without them. Performance with occlusions up to 30% is consistent with the reference test.

VI. CONCLUSION

In this paper, we have illustrated a low-cost stereo vision system and its application to object recognition. The hardware consists of a pair of consumer market cameras mounted on a rigid bar and costs less than 80 Euros. These cameras lack hardware-synchronized trigger signals and do not allow optics customization. In spite of such limitations, the point cloud obtained using the ROS packages for acquisition, calibration and disparity image computation is sufficiently accurate for the given task. The point cloud cluster containing the object to be recognized is identified under the hypothesis that such object lies on a planar surface and inside a given bounded region. The recognition algorithm is based on the extraction and comparison of FPFH features and is robust to partial views and to occlusions. Each candidate object is compared with the models contained into a dataset defined a priori. Experiments have been performed to assess the performance of the algorithm and have shown an overall recognition rate above 80%. The effect of occlusion on recognition rate has been assessed by showing that recognition performance is only slightly affected even when occlusion removes up to 30% of the object surface perceived from the current viewpoint.

In the system described in this work, the ROI is fixed and a single object is assumed to lie in the scene. We are currently working on an object detection algorithm dealing with less restrictive assumptions about the objects in the scene and the region of interest.

VII. ACKNOWLEDGEMENTS

We thank Elettric80 S.p.a. - Viano (Italy), for supporting this work.

REFERENCES

[1] J. Aleotti, D. Lodi Rizzini, and S. Caselli, "Object Categorization and Grasping by Parts from Range Scan Data," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2012, pp. 4190–4196.

[2] K. Konolige, "Projected texture stereo," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2010.

[3] M. Andersen, T. Jensen, P. Lisouski, A. Mortensen, M. Hansen, T. Gregersen, and P. Ahrendt, "Kinect depth sensor evaluation for computer vision applications," Technical report ECETR-6, Department of Engineering, Aarhus University (Denmark), Tech. Rep., 2012.

[4] D. Um, D. Ryu, and M. Kal, "Multiple intensity differentiation for 3-d surface reconstruction with mono-vision infrared proximity array sensor," *Sensors Journal, IEEE*, vol. 11, no. 12, pp. 3352–3358, 2011.

[5] G. Burel and H. Hènocq, "Three-dimensional invariants and their application to object recognition," *Signal Process.*, vol. 45, no. 1, pp. 1–22, 1995.

[6] A. Johnson, "Spin-images: A representation for 3-D surface matching," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, August 1997.

[7] T. Gatzke, C. Grimm, M. Garland, and S. Zelinka, "Curvature maps for local shape compariso," in *Proc. of Int. Conf. on Shape Modeling and Applications (SMI)*, 2005, pp. 246–255.

[8] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[9] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "Point feature extraction on 3D range scans taking into account object boundaries," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2011.

[10] R. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D registration," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2009, pp. 3212–3217.

[11] A. Aldoma, Z. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. Rusu, S. Gedikli, and M. Vincze, "Tutorial: Point cloud library: Three-dimensional object recognition and 6 DOF pose estimation," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 80–91, Sept. 2012.

[12] H. Kato and M. Billinghurst, "Marker tracking and HMD calibration for a video-based augmented reality conferencing system," in *Proc. of the Int. Workshop on Augmented Reality*, 1999.

[13] R. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Shanghai, China, May 9-13 2011.