World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:10, 2008

Recognition and Reconstruction of Partially Occluded Objects

Michela Lecca and Stefano Messelodi

***Abstract*—** A new automatic system for the recognition and reconstruction of rescaled and/or rotated partially occluded objects is presented. The objects to be recognized are described by 2D views and each view is occluded by several half-planes. The whole object views and their visible parts (*linear cuts*) are then stored in a database. To establish if a region R of an input image represents an object possibly occluded, the system generates a set of linear cuts of R and compare them with the elements in the database. Each linear cut of R is associated to the most similar database linear cut. R is recognized as an instance of the object O if the majority of the linear cuts of R are associated to a linear cut of views of O. In the case of recognition, the system reconstructs the occluded part of R and determines the scale factor and the orientation in the image plane of the recognized object view. The system has been tested on two different datasets of objects, showing good performance both in terms of recognition and reconstruction accuracy.

***Keywords*—** Occluded Object Recognition, Shape Reconstruction, Automatic Self-Adaptive Systems, Linear Cut.

I. INTRODUCTION

MANY industrial machine vision applications, like factory automation and quality control, and the challenges posed by Image Understanding made recognition of partially occluded objects one of the most investigated field in Computer Vision and Pattern Analysis. Given a region of an image, the problem is to determine whether it represents a portion of an object. The main difficulty is the definition of a model of the object, that permits its recognition from its parts. Moreover, since an object can appear under various conditions (e.g. rotated and/or rescaled, differently illuminated, deformed,...), the object model and/or the recognition strategy must be invariant with respect to these variations.

The most popular approaches describe the objects and the image portion by means of *local features*, like pixel intensity, corners, high curvature points, edge fragments, circle arcs approximating the boundary, inflections and bi-tangent points, [6], [12], [7], [15], [16], [9]. Other methods, e.g. [14], describe the objects and the image region by some subsets of pixels, allowing a stable detection and recognition of the objects. In other cases (e.g. [17], [13]), the description is done by dividing the object image into many regular parts, for example rectangles, and by describing each of them by global features. For objects consisting of distinguishable parts arranged in a fixed spatial configuration, the model is composed by a description of the parts along with information about their spatial relationships or even with a label about their functionality [10], [18]. Generally, in all these models, the local features are organized in hierarchical structures, for instance trees or graphs, and recognition is reduced to their matching [11], [7], [8].

The choice of the local features for the object description, their extraction from the image, and the matching algorithm for the recognition are often computationally expensive, requiring generally at least cubic time for the feature selection (e.g. for the Fourier transform computation) and even more for finding the feature correspondences, also when a priori knowledge or heuristic rules are introduced [19].

This work presents $(RO)^2$ (*R*ecognition and *R*econstruction of partially *O*ccluded *O*bjects), an automatic self-adaptive system that recognizes partially occluded objects and reconstructs their whole shape. Differently from the approaches enumerated above, the description of the database objects and of the image region to be recognized do not imply the use of local features. Moreover the computation of the descriptors as well as the features matching are very fast.

The main advantages in the use of $(RO)^2$ are the simplicity of the object model, the invariance by rescaling and rotating, and the restricted user interaction. The model of an object consists of many 2D views, representing the whole object from different points of view or an occlusion of it by a half-plane (*linear cut*). The recognition of a region R of a color image as an instance of an object possibly occluded is done by generating many linear cuts of R and by comparing them with those of each object. This process is guided by heuristic rules, that employ some thresholds related to geometric properties (area, orientation, scale factor) or visual similarity (L^1 distance and log-polar distance) of the linear cuts to compare. These heuristic parameters are dependent on the input database and on the half-planes used to build the object models. Since the system evaluates them in a complete automatic way, it is a self-adaptive system with respect to the input database and to the object models.

The main disadvantage is due to the fact that the system needs a large amount of memory space to store the database containing the objects and their cuts, but a distributed architecture can easily solve this problem.

Section II outlines $(RO)^2$; Section III illustrates the method for the construction of the object model and the technique for the automatic estimation of all the thresholds involved in the recognition process; Section IV explains the procedure for the recognition of occluded objects; Section V illustrates how the system reconstructs the whole shape of an occluded recognized object; Section VI presents some experiments and the obtained results; finally, Section VII contains some conclusions and the plans for the future work.

Manuscript received August 30, 2006. This work is supported by the European Project VIKEF (Virtual Information and Knowledge Environment Framework). M. Lecca and S. Messelodi are with ITC - irst - Centro per la Ricerca Scientifica e Tecnologica, 38050-Povo, Trento, Italy (e-mail: {lecca, messelod}@itc.it).

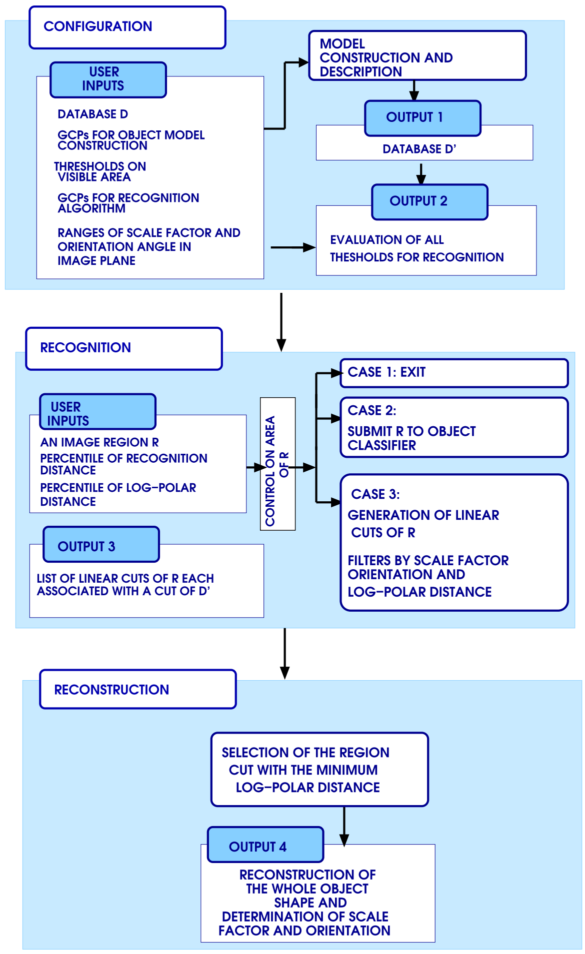

Fig. 1. $(RO)^2$ operating phases.

II. SYSTEM $(RO)^2$

System $(RO)^2$ manages databases in which the objects are represented by 2-dimensional views, as [1]. Let D be such a database. The system occludes each view by half-planes having different slope and masking different percentage of the view. Then it builds a new database D' containing the remaining visible parts (*linear cuts*) and D. To establish if a region R of an input image is a view possibly occluded of an object of D, the system generates a set of linear cuts of R and compares them with the items of D'. Each element of D', R, and each linear cut of it are described by means of a vector of low-level features, such as color, shape and texture. Visual similarity between a portion of R and an element of D' is defined as the L^1-distance between their correspondent feature vectors. The pairs (C_R, C_{O_v}), where C_R is a linear cut of R and C_{O_v} is the element of D' having the smallest distance from it (*recognition distance*), are stored in a list. R is recognized as an instance of an object O in D if the majority of the linear cuts of R is associated to a linear cut of views of O. The percentage of cuts of O associated to R is returned as *recognition confidence*. In case of recognition, the system selects the pair (C_R, C_{O_v}) such that the log-polar transforms of C_R and C_{O_v} match as well

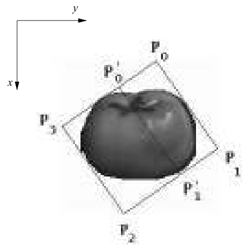

Fig. 2. Rectangles for the generation of the linear cuts of a view of an object of COIL-100 [1]. The direction θ of the occluding half-plane is P_0P_1 with respect to the reference system xy.

as possible, and uses O_v to reconstruct the whole shape of R. Since the descriptors are invariant by rescaling, rotation, and composition of thereof, the system is able to recognize also rotated and/or rescaled partially occluded views. The scale factor of R and its orientation in the image plane with respect to the associated object view are determined.

The key point of the method implemented in $(RO)^2$ is the concept of linear cut, that is formally defined as follows. Let Q be a planar region. Let be a half-plane intersecting Q. The sets $\cap Q$ and $Q - (\cap Q)$ are said the *linear cuts* of Q with respect to , or, briefly, the -linear cuts of Q.
The algorithm for the generation of $2h$ linear cuts of Q with respect to a sheaf of parallel half-planes $\{\ _i\}_{i=1,\ldots,h}$ with slope θ is based on the computation of the bounding rectangle $\mathbf{R}(\theta)$ of Q with direction θ (Fig. 2). Let P_0, P_1, P_2, P_3 be its vertices, and let $T = \{t_1, \ldots, t_h\}$ be a set of real values in (0, 1). For each i in $\{1, \ldots, h\}$, the system builds the rectangle $\mathbf{R}'(\theta)$ with vertices P_0', P_1', P_2, P_3, where

$$P_0' = P_0 + t_i(P_3 - P_0), \; P_1' = P_1 + t_i(P_3 - P_1),$$

and computes the $_i$-linear cuts

$$Q \cap P_0'P_1'P_2P_3, \qquad Q - (Q \cap P_0'P_1'P_2P_3).$$

The computation of the linear cuts of an object view O_v as well as of an input image region R is carried out by running the algorithm described above on a set $\{\theta_1, \ldots, \theta_k\}$ of slopes defined in $[0, \pi]$. In this work, these angles as well as the elements of T are equally spaced in their variability ranges. Therefore, we refer to the values k and h as the *generation cut parameters*, briefly indicated with GCPs.
The GCPs are set by the user. The GCPs used for the object model construction can be different from the GCPs used for the generation of the linear cuts of the region R. Moreover, the values of k and h can be different for each object in the database and even for each object view. A general prescription is to input a large number of linear cuts for constructing the model of the objects and a small number of cuts for the recognition procedure. However, the experiments reported in [20] showed that the system performances are stable with respect to changes of the GCPs used for the object model construction and in the recognition algorithm.

The work flow of $(RO)^2$ is shown in Fig. 1. It is partitioned in three main phases: *configuration*, *recognition*, and *reconstruction*.

III. Configuration

The Configuration phase consists of two steps: the object model construction and description, and the estimation of the thresholds used in the Recognition phase.

Object Model Construction and Description: The user inputs are an object database D, the GCPs n, m for the linear cut generation, and two real values p_a, p_b in (0,1), $p_a < p_b$, related to the percentage of the minimum visible area of the linear cuts. First, the system describes each object view O_v of D in terms of a vector of low-level feature, computes its area $A(O_v)$, occludes O_v by nm half-planes, and then inserts O_v and its linear cuts having area in $[p_a A(O_v), p_b A(O_v)]$ in a new database D'. The linear cuts of O_v with area greater than $p_b A(O_v)$ are described and compared with the items of D, and they are put in D' only if the object classifier does not recognize them as instances of O, i.e. the most similar item of D' returned by the object classifier is not a view of O.
The parameters p_a and p_b are used to avoid the generation of too small linear cuts and the insertion in D' of linear cuts very similar to the whole object view. Generally, p_a is close to 0, while p_b is close to 1.
The low-level feature vectors are computed by means of the extended version [4] of the content-base image retrieval system COMPASS [3] employed as object classifier. An image portion is described by means of histograms representing intensity, hue, saturation, edge distribution, while the Fourier coefficients and the Li Moments are used to describe its shape. The distributions of hue and saturation are represented also by two dimensional coocurrence matrices. The feature extraction and search process are very efficient [5], for example the description of an image with a size 128 × 128 pixels takes 0.13 seconds about, and retrieving the closest items in a database of 1 million elements takes less than a second on a standard Pentium4 2GHz CPU.
In this phase, the system computes the minimum and the maximum values A_{min} and A_{max} of the area of the elements of D'.

Estimation of Thresholds for Recognition: The user inputs of this steps are the GCPs n_R, m_R for the linear cuts of the input image region. Moreover, the user has to specify the ranges $[\alpha_1, \alpha_2]$, $[\psi_1, \psi_2]$ of variability for the scale factor and orientation in the image plane of the objects to be recognized. The main output of this phase is the estimate of all the thresholds employed in the recognition algorithm. These parameters are six: (a) the values $\alpha_1^2 A_{min}$, $\alpha_2^2 A_{max}$ regarding the minimum visible area of the cuts, and two thresholds $\langle E^A_{scale}\rangle$, $\langle E_{angle}\rangle$ related to the accuracy of the system on the determination of the scale factor and the orientation of rotated and/or rescaled object views or cuts (*geometric parameters*); (b) two thresholds and related to the object visual similarity (*similarity parameters*).

The estimation of $\langle E^A_{scale}\rangle$, $\langle E_{angle}\rangle$, and is carried out by two steps:

- *Object Transformation* - For each view O_v of an object O of D, the system computes a transformation $\varphi_{\alpha,\psi}$, given by the combination of the rescaling of factor α with the rotation of angle ψ in the image plane. α and ψ are randomly chosen in $[\alpha_1, \alpha_2]$, $[\psi_1, \psi_2]$ respectively. Let $\mathcal{S}$ be the set of the transformed object views of D.
  For each element of $\mathcal{S}$, the system generates at most $n_R m_R$ linear cuts having area greater than $\alpha_1^2 A_{min}$, and submits each of them to the object classifier, that describes and compares it with the elements of D': a linear cut C of $\varphi_{\alpha,\psi}(O_v)$ is *recognized* iff the item K of D' with the smallest distance $d(C)$ (*recognition distance*) from C is a view or a linear cut of a view of O. An element $\varphi_{\alpha,\phi}(O_v)$ of $\mathcal{S}$ is recognized iff the most of its linear cuts are associated to O.
- *Estimation of filtering and similarity thresholds* - For each cut generated in the previous step, $(RO)^2$ computes two estimates α_A and α_{lp} for the scale factor and an estimate ψ_{lp} for the orientation of C with respect to K. α_A is calculated as
$$\alpha_A = \sqrt{\frac{A(C)}{A(K)}} \tag{1}$$
  where $A()$ is the area of the region in the brackets. α_{lp} and ψ_{lp} are computed by means of the log-polar transforms $\mathcal{L}(C)$ and $\mathcal{L}(K)$. More precisely, $(RO)^2$ calculates the vertical and the horizontal shifts s_v and s_h of $\mathcal{L}(C)$ on $\mathcal{L}(K)$, such that $\mathcal{L}(C)$ and $\mathcal{L}(K)$ match as well as possible, i.e. such that their L^1 distance $\delta(C)$ in the normalized color space rgb is the smallest as possible.
  α_{lp} and ψ_{lp} are the anti-transformations of s_v and s_h respectively. The errors on the determination of scale factor and of the orientation are defined as follows:
$$E^A_{scale} = |\alpha - \alpha_A|, \; E^{lp}_{scale} = |\alpha - \alpha_{lp}|$$
$$E_{angle} = \min\{|\psi - \psi_{lp}|, 2\pi - |\psi - \psi_{lp}|\}.$$
  The system computes the mean values $\langle E^A_{scale}\rangle$ and $\langle E_{angle}\rangle$ and the distributions of the recognition distance and of the log-polar distance. The values of the threshold on the recognition distance of the linear cut, and of the threshold on the log-polar distance are fixed as the percentiles p_d and p_δ respectively, of these distributions.

The distributions of E^A_{scale} and E_{angle} are returned as measure of the system accuracy on the determination of the scale factor and the orientation. Moreover, $(RO)^2$ computes the percentages $\rho_C(n_R, m_R)$ of recognized cuts and the mean recognition confidence $\langle r_c(n_R, m_R)\rangle$ for the elements of $\mathcal{S}$. These values are an a-priori measure of the recognition system performances.

IV. Recognition

Let R be an input image region. The first step in the recognition procedure is a check on the area $A(R)$: (1) if $A(R) \notin [\alpha_1^2 A_{min}, \alpha_2^2 A_{max}]$, no recognition is made; (2) if $A(R) = \alpha_1^2 A_{min}$ the system compares the region R directly with the items of D; (3) if $A(R) \in (\alpha_1^2 A_{min}, \alpha_2^2 A_{max}]$, the system uses the method of linear cuts to recognize R (see Fig. 1).
In the third case (that we consider here), the recognition process consists of three phases:

Generation of linear cuts of R: the system computes the linear cuts of R with GCPs n_R, m_R. Only the linear cuts having area greater than $\alpha_1^2 A_{min}$ and scale factor (1) in $[\alpha_1, \alpha_2]$ are retained and described. Let C_R be such a cut.
The object classifier describes C_R in two ways: first, it uses all the descriptors and finds the most similar item K_R in D'; then the object classifier finds the most similar item K'_R in D' by excluding from the description the shape features. The control on the equality of K_R and K'_R is done to reduce possible errors of the object classifier. In fact, if K_R and K'_R differ, it is very probable that the cut C_R does not correspond to any cut or view of an object in D' (see Fig. 3). The output of this phase is a list L of pairs (C_R, K_R), with C_R such that K_R and K'_R are cuts of the same object, and the recognition distance $d(C_R)$ is smaller than .

Filter on scale factor: the system computes the *distribution of the weighted scale factor*. The weighted scale factor of the cut C_R is defined as $\alpha_A(C_R)w(C_R)$, where $\alpha_A(C_R)$ is the scale factor of C_R computed by (1), and

$$w(C_R) = \frac{\ - d(C_R)}{\ }. \tag{2}$$

In order to reduce the noise in the histogram of the weighted scale factors, the system computes the convolution $f * G$ between the distribution f of the weighted scale factors and a Gaussian kernel G with $\sigma = \langle E^A_{scale} \rangle$.
The system computes the *optimal scale factor* α_{opt} defined as the minimum value of the maximum points of the function $f * G$, and removes from L the pairs (C_R, K_R) such that $\alpha_A(C_R)$ does not belong to the range $[\alpha_{opt} - \langle E^A_{scale} \rangle, \alpha_{opt} + \langle E^A_{scale} \rangle]$.

Log-Polar Filter: for each element (C_R, K_R) of L the system estimates the scale factor $\alpha_{lp}(C_R)$ and the orientation $\psi_{lp}(C_R)$ in the image plane of C_R with respect to K_R by means of the log-polar transform. The estimate of the scale factor by this method is more precise than that given by (1), because it takes in account also the visual similarity between $\mathcal{L}(C_R)$ and $\mathcal{L}(K_R)$ and consequently between C_R and K_R.
The list L undergoes two filtering consecutive procedures:

a. the pairs (C_R, K_R) with $\delta(C_R)$ greater than the threshold are discarded;
b. the system computes the *distribution of the weighted orientation*, that is defined analogously to the distribution of the weighted scale factor. $(RO)^2$ computes the convolution $g * H$ of the distribution g of the weighted orientations with a Gaussian kernel H with $\sigma = \langle E_{angle} \rangle$. Then it computes the *optimal orientation* ψ_{opt} defined as the minimum value of the maximum points of the function $g * H$. Finally, it removes from L the pairs (C_R, K_R) such that $\psi_{lp}(C_R)$ does not belong to the range $[\psi_{opt} - \langle E_{angle} \rangle, \psi_{opt} + \langle E_{angle} \rangle]$.

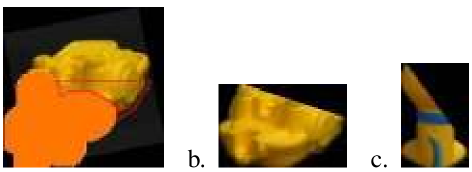

Fig. 3. For the linear cut highlighted in a., the items K'_R and K_R (in b. and c.) are different: this implies that the cut in a. is not reliable for the recognition.

If the list L is not empty, for each pair (C_R, K_R) in L the system specifies the object of D which K_R belongs to and returns the frequency of each object of D in the list L. R is then recognized as an instance of an object O iff the majority of its linear cuts are associated to a view, possibly occluded, of O. The recognition confidence is returned as measure of the recognition goodness.

V. Reconstruction

If R has been recognized as an instance of the object O of D, the system selects from the list L the pair (C_R, K_R) with the smallest log-polar distance $\delta(C_R)$. Let O_v be the object view whose K_R is a linear cut. The function η that maps K_R on C_R is given by the combination of the transformation

$$\phi(x) = \frac{1}{\alpha_{lp}(C_R)} \mathcal{R}_{-\psi_{lp}(C_R)} x, \quad x \in K_R \tag{3}$$

(where $\mathcal{R}_{-\psi_{lp}(C_R)}$ is the rotation of angle $-\psi_{lp}(C_R)$) with the translation by the vector $P = B(C_R) - B(\phi(K_R))$ (where $B(\cdot)$ denotes the barycenter).
The function $\eta : K_R \mapsto C_R$ is then extended to the function $\overline{\eta} : O_v \mapsto \mathbf{R^2}$, and $\overline{\eta}(O_v)$ is the reconstruction of the whole shape of R.

VI. Experiments

The recognition performance of $(RO)^2$ has been tested on the synthetic databases COIL-25 and IKEA-400, described in the following subsections. In both the cases, the extended database D' has been created by occluding the elements of the original database D by half-planes with slopes $\{\theta_1, \ldots, \theta_n\}$, such that

$$\theta_i = (i-1)\frac{\pi}{n}, \; i = 1, \ldots, n. \tag{4}$$

The set $T = \{t_1, \ldots, t_m\}$ is such that, considered the minimum bounding rectangles $\mathbf{R}(\theta_j)$ and $\mathbf{R}'(\theta_j)$ ($j \in \{1, \ldots, n\}$) defined in Section II, the euclidean distance $\parallel P_0 - P'_0 \parallel$ is x pixels. The GCP m is then a function of x, that by turns depends on the dimensions of the objects to be recognized. For instance, for elongated objects, x is small. Note that in the recognition algorithm it is not recommendable to use the information about the size of R, because in the case of occluded and rescaled objects, it may not correspond to the real size.

For both the databases, we set $p_a = 0.40$, $p_b = 0.90$, $[\alpha_1, \alpha_2] = [0.5, 1.5]$, $[\psi_1, \psi_2] = [0, 2\pi]$, and $p_d = p_\delta = 0.99$.

Experiments of the dependency of the recognition performance $(RO)^2$ on the GCPs used in the object model construction and in the recognition algorithm have been illustrated in [20] for the dataset COIL-25. They showed that the system performance is robust with respect to changes of the GCPs, and varies between the 96% and 100%.

To testing the recognition performance of the system, for each database, we construct four sets of images containing rescaled and/or rotated objects partially occluded. We denote these sets with S_{15}, S_{25}, S_{35}, S_{45}, where the subscript indicates the percentage of occluded area. The rescaling factors and the angles of rotation used for generating these sets have been randomly chosen in $[\alpha_1, \alpha_2]$, $[\psi_1, \psi_2]$ and they are different from those employed in the configuration phase for the construction of the set $\mathcal{S}$. Two examples of occluded objects of COIL-25 and IKEA-400 are shown in Figures 4. In all the cases the region R is the visible part of the object.

The reconstruction performances have been tested by estimating the *overlap index*

$$\nu := \frac{A(\overline{\eta}(O_v)) \cap \varphi_{\alpha,\psi}(O_v))}{A(\overline{\eta}(O_v) \cup \varphi_{\alpha,\psi}(O_v))}. \tag{5}$$

The computational time of the recognition algorithm varies from 27 seconds to 1 minute on a standard Pentium4 2.80 GHz. It depends mainly on the number of cuts of the region R submitted to the object classifier, and therefore on the GCPs (n_R, m_R). The description of R and the comparison of its cuts with the items of the database D' are not time-expensive thanks to the remarkable efficiency of the object classifier COMPASS. Moreover, the recognition algorithm can be easily parallelized, so that the computational times can be significantly reduced. The most expensive step is the computation of the log-polar transform, that is essential for the reconstruction, even thought its absence guaranties a fairly recognition of the occluded objects in a shorter time (from 7 to 19 seconds).

A. COIL-25

COIL-25 is a subset of the well known database COIL-100 [1], that consists of 100 objects, each one represented by 72 2D views. COIL-25 is composed by 25 objects of COIL-100 (those labeled from 1 to 25 in COIL-100) and the appearance of each object is represented by 12 views (one each 30 degrees). The number of images of COIL-25 is therefore 300.

The database COIL-25' containing the cuts of the elements of COIL-25 has been built by using the GCPs n = 10 and m such that x = 3 pixels. For these GCPs, the cardinality of COIL-25' is 58961. For the recognition algorithm, we fixed n_R = 10 and m_R = 10.

The thresholds estimated automatically have been reported in Table I-A, along with the recognition percentages $\rho_C(n_R, m_R)$ and the mean recognition confidence

a.  b. 

Fig. 4. Examples of test images. Each of them contains a rescaled and/or rotated object view of D partially occluded. a. contains an object of COIL-25 with occluded area percentage equal to 45%; b. is an object of IKEA-400, with occluded area percentages of 35%.

a. 

b. 

Fig. 5. Examples of shape reconstruction in the cases of object views of COIL-25 (a. with occlusion percentage 25%) and IKEA-400 (b. with occlusion percentage 35%).

$\langle r_C(n_R, m_R)\rangle$ estimated in the configuration phase. Table I-B shows the percentage of recognized occluded object views, the mean recognition confidence and the mean overlap index for each test set.

TABLE I

A. Values of Parameters estimated in Configuration Phase for COIL-25 (the areas are measured in pixels); B. recognition percentage, mean recognition confidence, and mean overlap index $\overline{\nu}$ for the test sets.

Table A.

Parameter	Value
Γ	0.0229
Δ	0.1979
$\alpha_1^2 A_{min}$	661.8
$\alpha_2^2 A_{max}$	22064
$\langle E_{scale}^A \rangle$	0.0368
$\langle E_{angle} \rangle$	0.1692
$\rho_C(10, 10)$	98.89%
$\langle r_C(10, 10)\rangle$	98.88 %

Table B.

Test Set	Recognized (%)	Confidence	$\overline{\nu}$
S_{15}	99.67	100.00	0.9132
S_{25}	98.67	99.33	0.8982
S_{35}	98.34	100.00	0.8616
S_{45}	98.34	99.89	0.8299

B. IKEA-400

Database IKEA-400 consists of 400 images of furniture (400 objects) of IKEA downloaded from the Internet catalog [2]. The extended database IKEA-400' has been constructed by setting n = 10 an x = 50, and it contains 10660 images. The GCPs used for the recognition process are n_R = 10 and m_R = 10. The parameters estimated in the configuration phase are shown in Table II-A. Table II-B shows also the recognition and reconstruction performances on the test sets.

TABLE II

A. VALUES OF PARAMETERS ESTIMATED IN CONFIGURATION PHASE FOR IKEA-400 (THE AREAS ARE MEASURED IN PIXELS); B. RECOGNITION PERCENTAGE, MEAN RECOGNITION CONFIDENCE, AND MEAN OVERLAP INDEX $\overline{\nu}$ FOR THE TEST SETS.

Table A.

Parameter	Value
Γ	0.036869
Δ	0.304194
$\alpha_1^2 A_{min}$	675.75
$\alpha_2^2 A_{max}$	438898.5
$\langle E^A_{scale} \rangle$	0.05117
$\langle E_{angle} \rangle$	0.442882
$\rho_C(10, 10)$	83.74%
$\langle r_C(10, 10) \rangle$	83.77%

Table B.

Test Set	Recognized (%)	Confidence	$\overline{\nu}$
S_{15}	99.00	99.85	0.7942
S_{25}	89.25	99.89	0.7756
S_{35}	96.75	99.90	0.7387
S_{45}	93.00	99.47	0.7300

Fig. 6. The object is occluded in more parts, so that the visible part does not contain a linear cut. In this case, the current recognition algorithm fails.

VII. CONCLUSIONS

The experiments illustrated in Section VI and [20] show $(RO)^2$ is a promising system for the automatic recognition and reconstruction of partially occluded object views.

Our future plans include the optimization of the algorithm for the computation of the log-polar transform and the code parallelization, to reduce the computational time.

Moreover, the current version of $(RO)^2$ is not able to recognize and reconstruct an object view when its boundary is occluded in more parts (Fig. 6), because the visible part does not contain a linear cut. So, our future works include the insertion in the object model of *multiple linear cuts*, i.e. simultaneous occlusions of each object view by more half-planes.

Other important tasks are the analysis of its performances on real-world images and the comparison of our system with others on a common dataset.

Finally, the strategy implemented in $(RO)^2$ will be integrated in the system MEMORI [21], to extend its capability to detect and recognize in color images the objects of a database also in presence of occlusions.

ACKNOWLEDGMENT

The authors acknowledge IKEA Italia SpA that made available for research purpose its catalogs. They also express their gratitude to Carla Maria Modena for the review of this paper.

REFERENCES

[1] S. A. Nene, S. K. Nayar, H. Murase, *Columbia object image library (COIL-100).* In *Technical Report CUCS-006-96, Columbia University*, 1996.

[2] http://www.ikea.com/ms/it_IT/our_products.html

[3] R. Brunelli, O. Mich, *Image retrieval by examples.* In *IEEE Transactions on Multimedia* N. 2(3), 2000.

[4] C. Andreatta, *CBIR techniques for object recognition. Technical Report* ITC-irst T04-12-01, December 2004.

[5] M. Lecca, *MEMORI - Version 1.0. Technical Report* ITC-irst T05-10-01, October 2005

[6] M. Reinhold, M. Grzegorzek, J. Denzler, H. Niemann, *Appearance-Based Recognition of 3-D Objects by Cluttered Background and Occlusions.* In *Pattern Recognition* Vol. 38, N. 5, 2005

[7] A. Shokoufandeh, I. Marsic, S. J. Dickinson, *View-based object recognition using saliency maps.* In *Image and Computing* N. 17, 1999

[8] V. Vilaplana, X. Giro, P.Salembier, F. Marques, *Region-based extraction and analysis of visual object information.* In *Proc. of Int. Workshop on Content-Based Multimedia Indexing CBMI 2005*, pp. SSI.3.1-SSI.3.9, SBN: 952-15-1364-0, 2005

[9] V. Ferrari, T. Tuytelaars, L. Van Gool, *Simultaneous Object Recognition and Segmentation by Image Exploration.* In *International Journal of Computer Vision (IJCV)*, April 2006

[10] E. Rivlin and S. J. Dickinson and A. Rosenfeld, *Recognition by Functional Parts.* In *Computer Vision and Image Understanding: CVIU*, Vol. 62, N. 2, 1995

[11] L. Wiskott, and C. von der Malsburg, *A Neural System for the Recognition of Partially Occluded Objects in Cluttered Scenes.* In *Advances in Pattern Recognition Systems using Neural Networks Technologies*, Vol. 7 in series Machine Perception and Artificial Intelligence, 1994

[12] F. Krolupper, *Recognition of Occluded Objects Using Curvature.* In Proc. of *The 12th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision: WSCG 2004*, 2004

[13] T. Deselaers, D. Keysers, R. Paredes, E. Vidal, H. Ney, *Local Representation for multi-object recognition*. In Proc. of *Deutsche Arbeitsgemeinschaft für Mustererkennung: DAGM 2003*, 2003

[14] S. Obdrzlek, J. Matas, *Object Recognition using Local Affi ne Frames on Distinguished Regions.* In Proc. of *British Machine Vision Conference*, 2002

[15] R. Basri, D. Jacobs, *Recognition Using Region Correspondences* In Proc. of *International Conference on Computer Vision: ICCV*, 1995

[16] G. Dorko, C. Schmid, *Object class recognition using discriminative local features.* In *Research Report 5497, INRIA Rhone Alpes*, 2004

[17] G. Fritz, L. Paletta, H. Bischof, *Object representation and recognition from informative local appearances.* In Proc. of *Digital Imaging in Media and Education*, 28th AAPR Workshop, 2004

[18] S. Agarwal, A. Awan, D. Roth, *Learning to Detect Objects in Images via a Sparse, Part-Based Representation.* In *IEEE, Pattern Analysis and Machine Intelligence*, Vol. 26, No. 11, 2004

[19] S. Berretti, A. Del Bimbo, E. Vicario, *Effi cient matching and indexing of Graph Models in Content-Based Retrieval.* In *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 10, 2001

[20] M. Lecca, *Recognition and Reconstruction of partially Occluded Objects.* Technical Report, ITC-irst T06-04-01, April 2006.

[21] M. Lecca, *Object Recognition in Color Images by the Self Confi guring System MEMORI.* In *International Journal of Signal Processing*, Vol. 3, No. 3, 2006