Grasping Unknown Objects using an Early Cognitive Vision System for General Scene Understanding

Mila Popović, Gert Kootstra, Jimmy Alison Jørgensen, Danica Kragic, Norbert Krüger

***Abstract*— Grasping unknown objects based on real-world visual input is a challenging problem. In this paper, we present an Early Cognitive Vision system that builds a hierarchical representation based on edge and texture information, which is a sparse but powerful description of the scene. Based on this representation we generate edge-based and surface-based grasps. The results show that the method generates successful grasps, that the edge and surface information are complementary, and that the method can deal with more complex scenes. We furthermore present a benchmark for visual-based grasping.**

I. INTRODUCTION

In this paper we propose a vision system for general scene understanding allowing for grasp planning. We focus on grasping unknown objects for which top-down knowledge cannot be applied easily. In contrast to 2D approaches, which often need simplifying assumptions on the actual action execution, e.g., [1], [2], we make use of 3D information in terms of contour and surface descriptors, allowing for improved grasp planning. In contrast to other 3D approaches that are based on segmenting scenes by different kinds of shape primitives, e.g., [3], [4], our approach does not require any kind of complex segmentation and registration process nor manual pre-processing, but operates on elements of a visually extracted hierarchical representation of the scene, which has not only been used for grasping, but for example for pose estimation and object recognition [5]. The presented method moreover deals with noise and uncertainty in the real world.

One of the problems in grasp planning is the nearly infinite number of possible grasps, which all need to be evaluated to assess their quality. Many current approaches therefore reduce the number of possible grasps by modeling the object shape with a number of shape primitives, such as boxes [3], cylinders, cones, spheres [4], or superquadrics [6]. With the approach we present, such explicit shape abstractions are not necessary. Our vision system inherently provides a sparse and abstract, but powerful set of 3D features. Making use of our hierarchical representation of the scene, the amount of computed grasps can be controlled by the granularity of the feature descriptors. Moreover, our 3D features are naturally aligned with the shape of the object, which is not necessary the case when shape primitives are used.

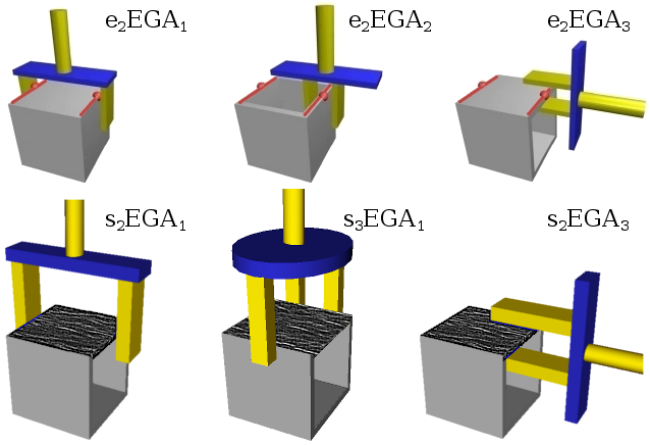

Fig. 1: The elementary grasping (EGA) actions. Top row: three types of edge-based EGAs. The red lines indicate the selected contours. Bottom row: three types of surface-based EGAs. The dark face shows the selected surface.

More specifically, we propose and evaluate a method for the bottom-up generation of two- and three-fingered grasps based on edge and surface structure. The edge and surface structures are extracted by means of an extension of the biologically-motivated hierarchical vision system [5]. This system, in the following called early cognitive vision (ECV) system, makes use of an elaborated mid-level ECV stage in which structurally rich and disambiguated information is provided to higher level of visual processing (for a detailed discussion see [7]). This system has been applied to the problem of grasping unknown objects [8] based on contour relations which are used to define so-called elementary grasping actions (EGAs) (see Fig. 1, top row). In this paper we extend the ECV system, which primarily was dealing with edge like structures [5], by texture information to allow for the association of grasps to surface information (see Fig 1, bottom row).

This perceptual organization process is guided by 2D and 3D relations defined between visual entities at the different

Mila Popović and Norbert Krüger are with the Cognitive Vision Lab, The Mærsk Mc-Kinney Møller Institute, University of Southern Denmark, Campusvej 55, DK-5230 Odense, Denmark {mila, norbert}@mmmi.sdu.dk

Gert Kootstra and Danica Kragic are with the Computer Vision and Active Perception Lab, CSC, Royal Institute of Technology (KTH), Stockholm, Sweden {kootstra, danik}@kth.se

Jimmy Alison Jørgensen is with the Robotics Lab, The Mærsk Mc-Kinney Møller Institute, University of Southern Denmark, Campusvej 55, DK-5230 Odense, Denmark jimali@mmmi.sdu.dk

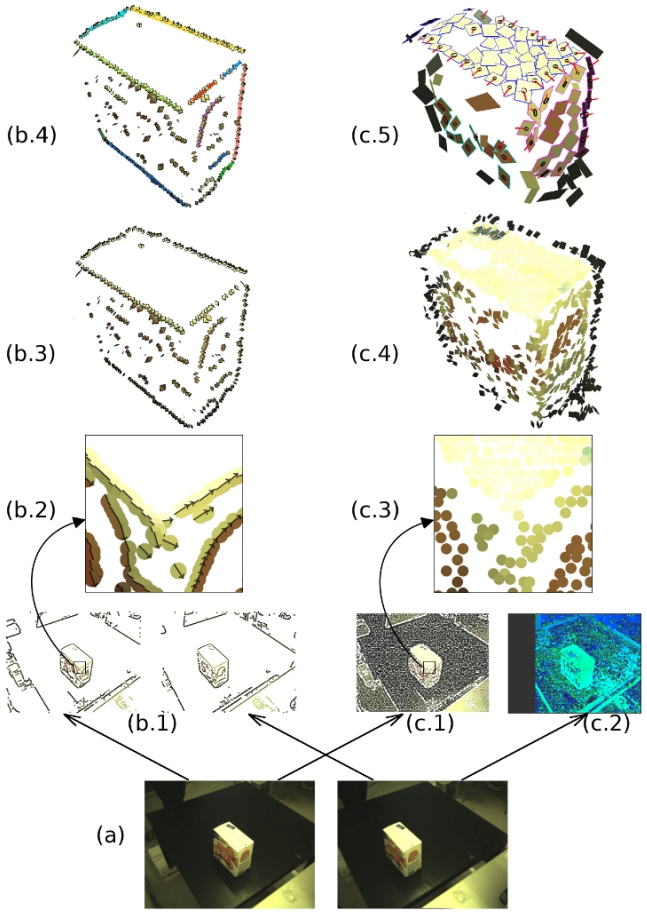

Fig. 2: The hierarchical representation of edge and texture information in the ECV system. (a) An example stereo image pair. (b.1) 2D line segments for the left and the right image. (b.2) a detail from b.1. (b.3) 3D line segments. (b.4) 3D contours. (c.1) 2D texlets for the left image. (c.2) disparity image. (c.3) a detail from c.1. (c.4) 3D texlets. (c.5) 3D surflings segmented into three surfaces, see also figure 4. This figure is best viewed in color.

levels of the hierarchy, namely local edge primitives and contours, as well as texlets, surflings and surfaces (see Fig. 2). These relations in particular allow for the extraction of orientation and depth discontinuities which then can be used for efficient surface segmentation. Once the 3D surfaces with their boundaries have been extracted, we can associate a set of grasping hypotheses to a single boundary primitive or to doublets or triplets of boundary primitives (see Fig. 4). These generated grasps are then evaluated and ranked by different visual quality measures.

We do systematic tests of these hypotheses in a mixed real-world and simulated environment in which features are extracted from real visual data and grasps are performed in a virtual environment using a dynamic simulation (see Fig. 3). This allows us to test a large number of grasps (in total over 30,000 grasps were tested) generated from natural stereo images. By that we can make elaborated quantifications of edge-based and surface-based grasps and their associated visual quality measurements

This paper has the following contributions: (1) We extend the ECV system with a hierarchy of features in the texture domain, (2) the proposed features give a sparse and abstract, but meaningful representation of the scene, which on one side reduces the search space for grasping and on the other side creates additional context information which is relevant for grasping, (3) we show the complementary strength of edge and texture information for grasping, and (4) we present a benchmark for vision-based grasping.

II. Related Work on Vision-based Grasping

Different approaches to visual-based object grasping have been proposed by the robotic community. As proposed in [9], these approaches can be roughly divided into grasping of *known*, *familiar*, and *unknown* objects.

In grasping known objects, a detailed 2D or 3D model of the object is generally available. This model is then fitted to the current visual observation to retrieve the pose of the object. Based on the model and the pose estimation, a large number of grasps suggestions are generated and their quality is evaluated to select the most promising grasp, e.g., [10], [11], [12]. One of the main challenges is the huge amount of possible grasps. In order to reduce the search space, different techniques have been applied. In [4], [6], [3], the shape of the object is simplified by using shape primitives, such as, spheres, boxes, and superquadrics, thereby reducing the number of possible grasps. A dimensionality reduction of the configuration space of the hand using so-called eigengrasps has been proposed in [13]. It has been demonstrated in [14] that a small random subsample of the possible grasps is enough if not the best, but a good-enough grasp is sufficient.

The above-mentioned studies have been done in simulation, assuming complete knowledge about the object and the robot, and ignoring noise, with the exception of [3], where incomplete and noisy data has been used as well. The studies all assume a perfect segmentation of the object from its background. In contrast, we propose a method based on real visual data without any knowledge about the presented objects.

In the grasping of familiar objects, the system is generally trained on a set of objects and learns the relation between some visual features and the grasp quality. This knowledge is then used to grasp novel objects. In [15], for instance, the graspability of object parts is learned based on the parameters of superquadrics fitted to segmented parts of the object and using human expertise. A SVM has been trained to predict the grasp quality based on the hand configuration and the parameters of a single-superquadric representation of the objects in [16]. In [17], grasp knowledge is learned on a set of simple geometrical shapes and applied to grasp novel objects. All these experiments were done completely in simulation with synthesized data. In [1] grasping is learned based on a set of local 2D images features using synthesized objects, and this knowledge is used to grasp objects in the real world. The feature vector is high dimensional set of edge, texture and color features on different scales. Different features of two edge points resulting from our ECV system have been used in [18] to learn to predict the grasping success.

When grasping unknown objects, no model of the objects or prior grasp knowledge is used and all reasoning about

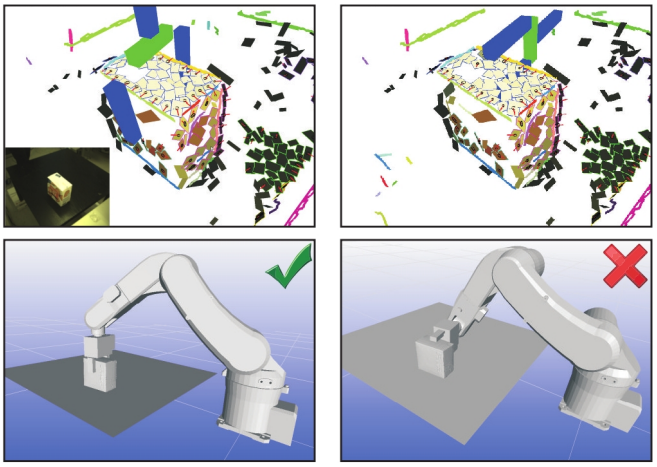

Fig. 3: Examples of two different grasps. The grasps are generated based on the ECV representation of the real scene (first row). The grasps are then tested in a simulated environment (second row). The grasp in the first column results in a successful grasp, whereas the grasp in the second column fails.

grasping is done on the visual observation of the scene. In [3] and [19], shape primitives, respectively boxes and quadrics, are used to deal with the noisy and incomplete data coming from robotic sensors, and to provide a reduced set of potential grasps. A sophisticated 3D representation of the scene based on our ECV system was used in [20], [8] for grasp planning. We build upon this work by extending the system to not only take edge features into consideration, but also texture features.

Most of the studies on vision-based grasping assume a segmentation of the objects from their background. However, for grasping unknown objects in real-world situations this assumption does not hold. When using pinch grasps, e.g., [1], single image points are sufficient, making object segmentation to relate multiple points on the same object unnecessary. Input of the user is taken to initialize object segmentation using active contours in [19]. In [21], a bottom-up segmentation method based on color and depth information is used and the graspability of the segments is learned using an SVM. In [8], we associated two grasp points with the same surface of an object by using coplanarity and cocolority.

In this paper, we present bottom-up visual methods for grasping unknown objects, based on unsegmented real-world scenes. Unlike other approached discussed in this section, we do not use a simplification of the object(s) using shape primitives to abstract the shape. Instead we extend the ECV system to produce a sparse, yet semantically meaningful representation of the scene that remains close to the true shapes of the objects and which allows the system to utilize the potential of edge as well as texture information.

III. THEORETICAL FRAMEWORK

A. ECV System

The framework of the Early Cognitive Vision system provides a rich visual representation that includes edges, textured surfaces and junctions [7]. The representation is layered and starts with extracting sparse local features from 2D images and categorizing them into one of the three categories mentioned above. These basic features are called *multi-modal primitives* and code both geometric and appearance information. By matching 2D features over two stereo views, the system derives corresponding 3D descriptors for different structures.

On the second level, ECV organizes basic features into perceptual groups (in both 2D and 3D) [22]. Edge segments are grouped into contours, while textured surface patches are organized in so-called *surflings*. On this higher level of abstraction it is again possible to group complex features or observe their relations. For example one can observe co-planar or co-colored contours, or combine surflings that are proximate in position and orientation into *surfaces*.

Figure 2 shows different stages in creating the hierarchical representation. The left-hand branch illustrates the propagation of edge information, and the right-hand branch shows the propagation of texture information.

1) Edges, Contours: Line segments in ECV are local edge feature descriptors (see Fig. 2b.2) that integrate geometrical (position, orientation) and appearance (color, phase) information. The local edge features are grouped into bigger perceptual groups – contours – based on multi-modal constraints including proximity, collinearity, co-circularity, and similarity in appearance. (see Fig. 2b.4). Contours, as features on the higher level of abstraction, can again be compared and grouped by observing relations between them. Relations of co-planarity and co-colority have for instance been used to trigger edge-based grasping actions (see [22], [8] and section III-B).

2) Texlets, Surflings, Surfaces: Texlets describe local properties of a textured surface. They store mean color, position and orientation of a surface patch. Arrangement of texlets into surfaces is done in two steps. Texlets are firstly combined into semi-global surface descriptors called surflings and subsequently into surfaces.

Initial texlets in 2D are extracted from the left image of the stereo pair (see Fig. 2c.3). The image plane is separated into local patches by applying a hexagonal grid. Texlet features are defined by averaging appearance and disparity information over pixels in a grid. Due to the grid sampling, each texlet in 2D will have up to six neighbor texlets. After constructing texlets in 3D, the neighboring structure is propagated from 2D to 3D (see Fig. 2c.4). Only texlets that remain close by in 3D space will remain neighbors.

In the next level of the hierarchy, the texlets are grouped into surflings. By looking at the similarity in color between neighboring texlets in 3D and by using the relation of transitivity, all texlets in a scene are grouped into pools sharing similar properties in color, position and 3D orientation. When using transitivity to associate texlets over long distances, it is possible that gradual changes will lead to non-optimal grouping. The system therefore subdivides the pools of texlets into small subsets of about five to ten texlets using k-means clustering on the position. These subsets of similar texlets form the surflings, which are semi-global surface features (see Fig. 2c.5). The geometrical information of a

surfling is obtained by fitting a plane through the underlying texlet positions, and the appearance information by averaging over the texlets. A surfling feature is described by a full 6D pose, length, width, and mean color.

Surfaces in the ECV system are constructed from surflings, in a similar fashion as surflings are from texlets (see Figs. 2c.5 and 4). The system establishes a neighboring structure between surflings with proximate position and orientation. In the formation of surfaces from surflings, color is not considered for grouping, but only position and orientation information. This allows for the representation of heterogeneously colored surfaces. Once surface features are in place, the system can label surflings that are at the boundaries of the specific surfaces.

This hierarchical visual representation provides an abstract description of the scene in terms of contours and surfaces. In the next section, we apply this general scene representation to the problem of grasping unknown objects. By matching contours and finding surfaces of objects, we can discarding many inadequate grasps and thereby drastically reduce the search space.

B. Grasp Generation

In this paper we look at two methods for generating *Elementary Grasping Actions* (EGAs). The first method generates two-fingered grasps based on *edge* features (e_2EGAs), while the second method uses *surface* features to generate two-fingered and three-finger grasps (s_2EGAs, s_3EGAs), see Fig. 1. We assume that the object is in the field of view of the camera and within the robot's reach.

In the case of both eEGAs and sEGAs we want to generate grasps that relate to the same surfaces of the object, which our ECV vision system allows for. Without any top-down information about the object, it is challenging to identify surfaces that belong to the objects and to find contact points that are located at the same surface of the object. The system does not distinguish foreground/background features, but limits the search for graspable features to the space above the known table plane. Moreover orientation information is needed to generate the grasp. Here we propose different methods to achieve this based on edge and surface information. For edge-based grasping, edge contours are matched by looking at their coplanarity and cocolority. The contact points are placed at the centers of the matched contours, while contact normals are calculated based on the orientation of the common surface between the contours and the individual contour orientations (see Fig. 1). For surface-based grasping, the elementary surflings are grouped based on position and orientation to form surfaces as discussed in the previous section. Surflings at the boundary of a surface become potential grasp points, while contact normals are computed based on the orientations of the individual boundary surflings and the orientation of a surface fitted through the boundary surflings.

In case of the $e_2\text{EGA}_1$ and the $s_{2,3}\text{EGA}_1$ encompassing grasps, the assumed surface is placed between the gripper's fingers, where the orientation of the gripper is perpendicular

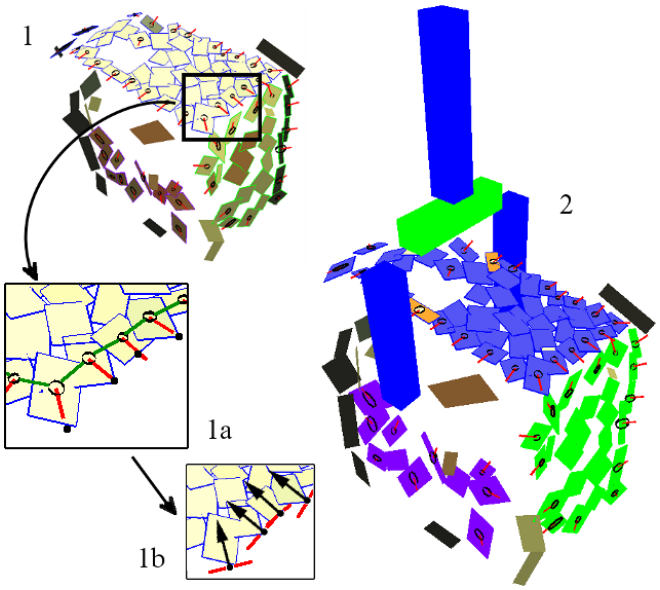

Fig. 4: (1) Examples of boundary surflings. (1a) The boundary is shown by the green line, the boundary-surfling direction by the red line, and the contact point by the black dot. (1b) The corresponding contact points are given by the black dot, with the black arrow showing the contact direction, and the red line showing the contact region. (2) The surface segmentation has been made more explicit for the purpose of illustration. An example two-fingered grasping action is shown, which is triggered by two boundary surflings of the top surface, highlighted in orange.

to the surface running through the predicted contact points. For the $e_2\text{EGA}_2$ pinch grasp, the same orientation is used, but the gripper attempts to grasp one of the edges, assuming empty space between the edges. The gripper approaches the surface from the side for the $e_2\text{EGA}_3$ and the $s_2\text{EGA}_3$ pinch grasps, with the approach direction aligned with the vector that lies in the surface and is normal to the local edge orientation. See Fig. 1 for an illustration.

Edge Elementary Grasping Actions (e_2EGAs) are built upon 3D edge features in a scene. The basic idea is that contours originating from the same object and from the same surface often share properties. Such contours will for instance often be co-planar and co-colored. The method presented in [8] first finds pairs of such contours and then generates a number of grasping actions for those contour pairs. Grasping actions are divided into three types and illustrated in Fig. 1, top row. For details we refer to [8].

The main contribution of this paper is in surface extraction and surface-based grasping. The *Surface Elementary Grasping Actions* ($s_{2,3}$EGAs) are described in the next section.

C. Surface Elementary Grasping Actions

The textural hierarchy presented in Section III-A identifies surfaces of the objects in the scene. This information is used for *Surface Elementary Grasping Actions* ($s_{2,3}$EGAs). The grasping actions belong to one of the two types shown on Fig. 1, bottom row; the $s_{2,3}\text{EGA}_1$ encompassing grasp and the $s_2\text{EGA}_3$ pinch grasp. The procedure of creating sEGAs involves two steps: the extraction and the selection of contact points.

Step 1: Contact points extraction: Surfaces in the ECV system are collections of surflings. Surflings within one

surface are labeled as boundary and non-boundary. Boundary surflings have a direction, which is aligned with the surface, pointing outwards, perpendicular to the boundary orientation (see Fig. 4(1a)). The computation of grasping actions is based only on boundary surflings. To make reasoning about the grasps more convenient, a plane is fitted through the boundary surflings and the surflings' positions and directions are projected on this plane.

For each non-corner boundary surfling a contact point is created. The contact point is positioned at the outer edge of the surfling, as illustrated in Fig. 4(1a). The contact normal is defined as the inverse direction of the projected surfling's direction and the region of a contact point is determined by the length along the boundary (see Fig. 4(1b)).

If a surface has a narrow area with only one surfling in width, the surfling will support two boundaries and contact points will be created on each side.

For the $s_{2,3}\text{EGA}_1$ encompassing grasps, pairs or triplets of boundary surflings are selected and the contact points are determined as described above. In the case of $s_2\text{EGA}_3$ pinch grasps, contact points are created above and bellow each boundary surflings, such that the gripper fingers are forming an opposition closing on the selected boundary surfling.

Step 2: Contact points selection: The next step is to select suitable contact points for the different grasp types. In case of $s_2\text{EGA}_3$ pinch grasps, each contact point will trigger a valid grasping attempt. In cases of $s_2\text{EGA}_1$ and $s_3\text{EGA}_1$, we are looking into combinations of contacts, where only some of them will represent a valid grasp.

By assuming a planar placement of contact points in 3D, we can utilize considerations similar to [23], but without constraints that purely 2D methods put on a grasp. The assumption thus made is that the shape of the object in the very vicinity of the visible surface preserves the shape outlined by the surface boundary. This is a less constrained assumption then the common assumption made in the purely 2D grasp approaches, where an outline of the whole object, given from a single 2D view, is used to reason about grasp properties, even though it is not sure at which depths different boundary extremities arise, nor at which depth the object ends.

The selection criteria are based on visual features and gripper-specific kinematic constraints. Grasp-stability measures based on wrench space are often used [24]. However, these measures usually assume perfect knowledge of the object's shape. Since we are dealing with visual observations of unknown objects in the real world, the derived shape representation will be somewhat noisy and uncertain. We therefore apply less detailed heuristics instead, with the purpose to maximize the grasp stability.

Encompassing $s_2\text{EGA}_1$ and $s_3\text{EGA}_1$ grasps are based upon pairs and triplets of contact points. In order to quickly reduce the number of contacts combinations, the constraints are given in the order of increased computational complexity. The first constraint filters out the contact combinations that are too far apart, having in mind the maximal distance between the fingers. In order to prevent sliding and to minimize the torques, we apply the second constraint, where the Coulomb's friction model is used: $F_f \leq \mu \cdot F_n$. The friction coefficient μ defines a friction cone half angle: $\beta = \arctan(\mu)$, that is the maximum distance in angle between the contact normal and the direction of the force applied by the finger, at which sliding will not occur. Since the friction coefficient is not known, we use a constant value.

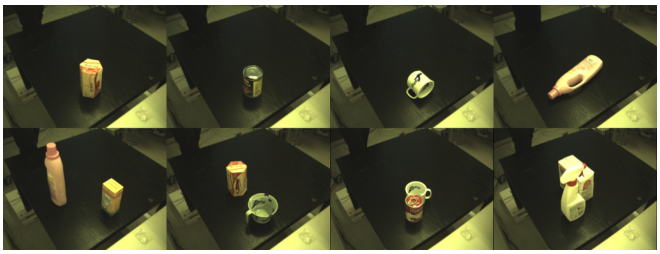

Fig. 5: Experimental scenes, left camera images. Top row: single object scenes. In the first two images (from left to right) objects are the in upright position, in the last two images objects are laying down. Bottom row: scenes with two objects. In the first two images objects are placed apart from each other, in the last two images they are close by.

In case of contact pairs we require that the angle between the contact normals is within $\pi \pm \beta$, where the contact normals point towards each other. Additionally we project contact regions in the direction of the contact normals, and demand that at least one of the projected regions intersects with the opposite contact's region.

For triplets of contacts $s_3\text{EGA}_1$, we require that the contact normals positively span the plane and that the intersection of friction cones is not empty [25]. The three vectors are said to positively span a plane, if each one of them can be written as a positive combination of the other two.

As described in section III-C, the contact points for the pinch grasps $s_2\text{EGA}_3$ are constructed so that one is above and the other bellow the selected boundary surflings. All such contacts pairs are sources for valid pinch side grasps, and no constraints are used for the selection.

IV. EXPERIMENTS

We used a mixed real-world and simulated experimental setup to test the proposed method, see Fig 3. Real camera images provide the input to the ECV system and the grasp-generation method. The produced grasp hypotheses are then tested in a dynamic simulated environment. This setup gives us the possibility to run a large number of trials and to repeat the experiments in the exact same conditions, while having to deal with the noise and uncertainty of the real world.

We made the stereo-image database, the simulated environment, and the dynamic simulator available for the public, so that it can be used as a visual-grasping benchmark[1].

A. The Real-World Visual Input

We have recorded a large number of stereo images of scenes with single and double objects. For the single-object scenes, stereo images have been taken from 11 different objects, each in 16 different poses. The double-object scenes

[1]http://www.csc.kth.se/~kootstra/visualGraspingBenchmark

contain 5 sets of two objects, each in 16 different configurations. Examples of the scenes can be appreciated in Fig. 5[1]. Based on these real images, grasps have been generated using the method outlined in Section III. In total around 18,000 grasps have been generated for the single-object scenes and 17,000 for the double-object scenes.

B. Grasping in Simulation

The grasps are performed in simulation using RobWork[2] (see Fig. 3). RobWork is a dynamic simulator for robotic grasping, which has been used in several related experiments [26]. The simulator has been shown to be very realistic in [27], where several thousands of grasps with a parallel gripper in a real robotic setup have been compared to the simulation. Using a dynamic simulator allows us to not only look at static quality measures of the grasp, but also to determine the actual grasp success by observing the dynamical and physical consequences of the grasp.

The objects that we used are included in the KIT Object-Models Web Database[3], which includes 3D models of the objects. These models have been obtained using a laser scanner and therefore provide a realistic representation of the scene. The object models have been placed at the correct positions in the simulation by registering them with the 3D point clouds obtained using OpenCV's stereo-matching algorithms. Besides the objects, also the table has been placed in the simulation. We would like to stress that these models have not been used for the generation of grasps.

In the experiments, we use a simulation of the Schunk dexterous hand (SDH-2), which is a three-fingered hand that can also be used for two-fingered parallel grasps. We use both options. Since we focus on the grasping of objects and the quality of the selected contact points, we do not simulate the robotic arm. The hand can therefore freely move around.

The desired configuration of the hand is determined based on the predicted contact points and the EGA type using inverse kinematics based on the position of the contact points and the desired approach direction of the gripper as explained in Sec. III. The hand is placed so that the finger tips go 2 cm past the surface in order to get a stable grip on the object. During the simulation, the hand is initially opened a bit further than the configuration based on the predicted contact points, and is then closed until a stable grasp is reached or the grasp fails. If a stable grasp is reached, the robot attempts to lift the object.

C. Evaluation of the grasps

Because we test the grasps using the dynamic simulator, we don't have to rely on static grasp quality measures, such as the grasp wrench space, but we can actually observe the dynamic consequences of grasping and lifting the object.

We determine the grasp stability by the lift result, q_{lift}. The lift results is a value between 0.0 and 1.0, which is inversely related to how much the object moves with respect to the hand during lifting, that is $q_{\text{lift}} = 1 - ||\mathbf{h} - \mathbf{o}||/||\mathbf{h}||$, where $\mathbf{h}$ is the vector of the displacement of the hand during lifting, and $\mathbf{o}$ that of the object. The following discretization can be made: *no slip* if $q_{\text{lift}} \geq 0.9$, *low slip* if $0.5 \leq q_{\text{lift}} < 0.9$, *high slip* if $0.2 \leq q_{\text{lift}} < 0.5$, and *drop* if $q_{\text{lift}} < 0.2$. In case that the grasp already fails before the lift, the status is *miss*, and $q_{\text{lift}} = 0$. In the experiments, we speak of *success* when there is no slip or low slip. The grasps where the generated hand configuration is in initial collision with the objects or table in the scene are not considered, since it would be straightforward to detect based on our visual scene representation.

[2]http://www.robwork.org

[3]http://wwwiaim.ira.uka.de/ObjectModels

V. RESULTS

Figures 6 and 7 show the success rates for the different grasp types as a function of the number of executed grasps, N. For a given scene, N grasps are taken at random from the set of generated grasps. There is success if any of these grasps is successful. The combined success rates are shown as well, where the N grasps of the six different grasp types are taken together. The plots show the average success rate for all scenes over 20 runs.

Figure 6 shows the results for the single-object scenes. The overall results are split up for scenes with the objects in upright position and for scenes with the object lying down. For the scenes with standing objects, the surface-based grasps outperform the edge-based grasps, with the three-fingered encompassing grasp ($s_3\text{EGA}_1$) as most successful. The edge-based pinch grasp $e_2\text{EGA}_2$ performs worst. It is clearly visible from the combined grasps that the different grasp types are complementary. The scenes with the object lying down show to be more challenging. With the exception of the $s_3\text{EGA}_1$, the success rates are low. Again, the combined grasps greatly improve the results.

The grasp success rates for the double-object scenes are given in Fig. 7. The two- and three-fingered surface-based grasps ($s_{2,3}\text{EGA}_1$) perform very good in the scenes where the objects are not touching. $e_2\text{EGA}_2$ has the lowest success rate. Again the grasp types show to be complementary with an excellent combined result. The scenes where the objects are touching are more difficult. The reason is that for some of the scenes, the grasp types generate non or only few collision-free grasps.

Overall, the surface-based grasps show higher success rates than the edge-based grasps. Comparing the two-fingered with the three-fingered surface-based grasps, we can see that having three fingers greatly improves the grasp stability. If we look at the overall results of the single-object scenes and the double-object scenes, we see that the success rates for the combined grasps are in the same range, despite growing visual complexity.

Figure 8 gives the success rate of the combined grasps per object for $N = 1$. For the single-object scenes, the results are quite variable for the different objects. In general it can be observed that wider cylindrical objects are more difficult to grasp. This can be explained because the fingers are sometimes positioned not deep enough on the objects so that the angle with the surface makes the object slips

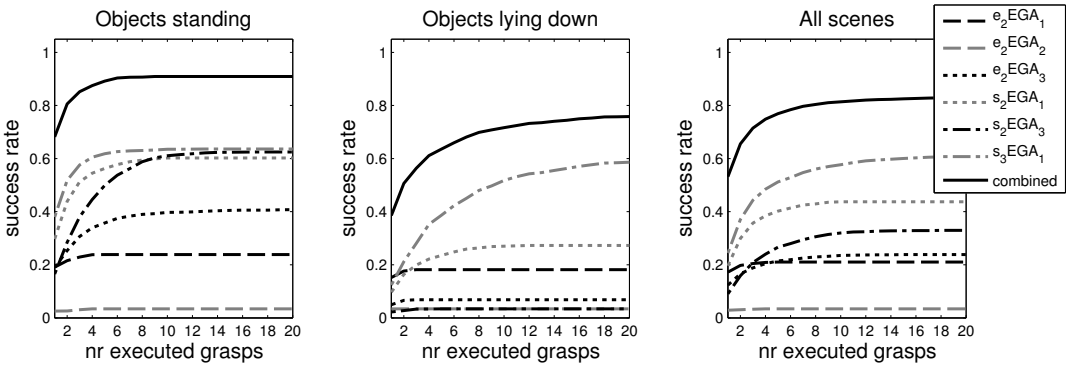

Fig. 6: Grasp success of the different grasp types for the single-object scenes. The success rate is plotted as a function of the number of executed grasps (N). A grasp is successful if any of the attempted grasps succeeds. Note that for the complementary results each of the different grasp types is attempted N times. The plots are the mean over 20 runs.

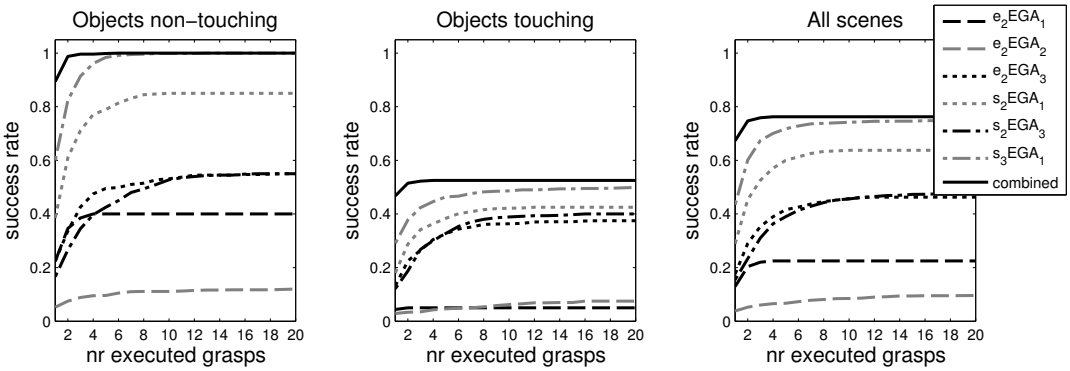

Fig. 7: Grasp success of the different grasp types for the double-object scenes.

out of the gripper. Top grasps are often successful on the standing objects, but are not possible for the objects lying down because of the size, which explains the lower success rates for lying objects. For double-object scenes with the two objects not touching each other, the success rates are consistently high. When the two objects touch each other, success rates are particularly low for those objects that are similar in height, leaving less space for a collision-free grasp. In general it seems that the graspability of an object improves if it is better textured.

Given the sparse representations of the scene and the heuristics for grasp selection, the grasp-generation method suggests only a small number of grasps. On average 20-40 grasps are generated per scene for the e_2EGAs, the s_2EGAs and the s_3EGAs. And as can be seen in Figs. 6 and 7, good grasp results are generally achieved already after a few attempts.

VI. DISCUSSION

In this paper, we presented a bottom-up vision system for general scene understand and used it for grasping unknown objects. We continued our earlier work [8], by extending the hierarchical representation of the Early Cognitive Vision system to the texture and surface domain, and by using the representation to generate not only edge-based, but also surface-based grasps. The ECV system organizes multi-modal visual information with growing levels of abstraction. This approach has two advantages: 1) the process of abstracting narrows the search space for grasping, and 2) contextual knowledge is added that allows to extract contact points on the same surface of an object. We furthermore presented a mixed real-world and simulated experimental setup. Based on real stereo images, our method builds a visual representation of the scene and generates grasps. These grasps are then tested in a dynamic simulator. This setup allowed us to test a large number of grasps and get quantitative results, while still dealing with the noise and uncertainty in the real-world visual data.

The results show that the edge-based and surface-based grasps complement each other. While surface-based grasps showed better success rates overall, the edge-based method, for instance, has the advantage that it can generate grasps also on low-textured objects.

If we compare the overall results of the single-object scenes with the double-object scenes, we see that the success rates for the combined grasps are in the same range. This shows that our hierarchical vision system makes a powerful representation of the scene that can be used to generate good grasps, even with increasing visual complexity.

Since we are dealing with grasping unknown object in unknown scenes, a 100% success is not expected. However, the results show that the grasp success strongly increases if more attempts are made. Using developmental learning mechanisms, such as we have proposed in [20], our future system can learn to improve from these explorations over time.

The benchmark that we have presented in this paper is open for scientific use. The stereo images, the simulated

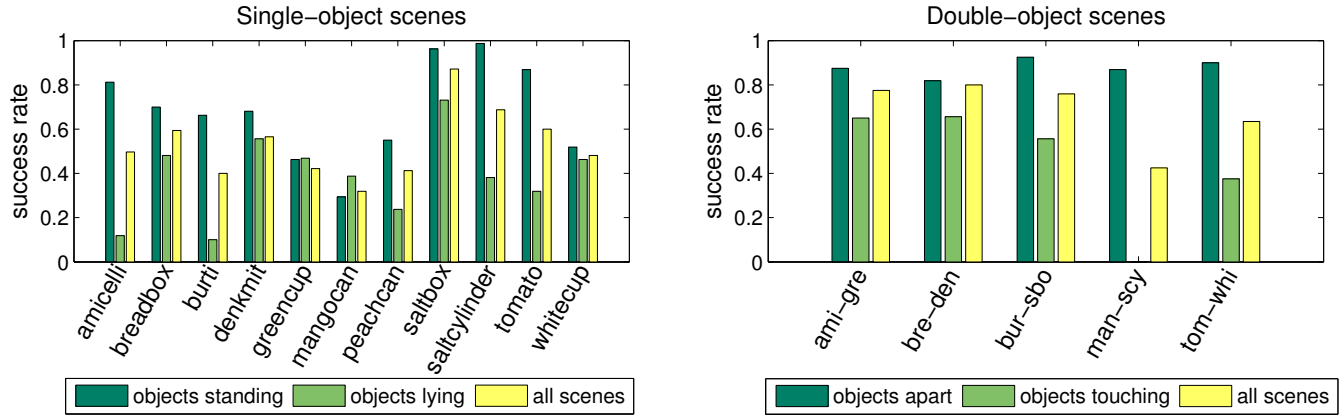

Fig. 8: The combined-grasp success rates per object. One grasp of each of the six grasp type is attempted. For the single-object scenes, the results are shown for the objects standing up and lying down, whereas for the double-object scenes, the results are given for the scenes with the objects apart and touching.

environment, and the dynamic simulator are available on http://www.csc.kth.se/~kootstra/visualGraspingBenchmark

VII. Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement n°270273, from the EU project eSMCs (IST-FP7-IP-270212), and from The Danish National Advanced Technology Foundation through the project Bin Picker.

References

[1] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic grasping of novel objects using vision," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.

[2] E. Chinellato, A. Morales, R. B. Fisher, and A. P. del Pobil, "Visual quality measures for characterizing planar robot grasps," *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, vol. 35, no. 1, pp. 30–41, 2005.

[3] K. Hübner, S. Ruthotto, and D. Kragic, "Minimum volume bounding box decomposition for shape approximation in robot grasping," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'08)*, 2008, pp. 1628–1633.

[4] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'03)*, 2003, pp. 1824–1829.

[5] N. Pugeault, F. Wörgötter, and N. Krüger, "Visual primitives: Local, condensed, and semantically rich visual descriptors and their applications in robotics," *International Journal of Humanoid Robotics (Special Issue on Cognitive Humanoid Vision)*, vol. 7, no. 3, pp. 379–405, 2010.

[6] C. Goldfeder, P. K. Allen, C. Lackner, and R. Pelossof, "Grasp planning via decomposition trees," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'07)*, 2007.

[7] N. Krger, N. Pugeault, E. Baeski, L. B. W. Jensen, S. Kalkan, D. Kraft, J. B. Jessen, F. Pilz, A. K. Nielsen, M. Popovi, T. Asfour, J. Piater, D. Kragic, and F. Wrgtter., "Early cognitive vision as a front-end for cognitive systems." *ECCV 2010 Workshop on "Vision for Cognitive Tasks"*, 2010.

[8] M. Popović, D. Kraft, L. Bodenhagen, E. Başeski, N. Pugeault, D. Kragic, T. Asfour, and N. Krüger, "A strategy for grasping unknown objects based on co-planarity and colour information," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 551 – 565, 2010.

[9] J. Bohg and D. Kragic, "Learning grasping points with shape context," *Robotics and Autonomous Systems*, vol. 58, no. 4, pp. 362–377, 2010.

[10] V. Nguyen, "Constructing stable grasps," *International Journal on Robotic Research*, vol. 8, no. 1, pp. 26–37, 1989.

[11] K. Shimoga, "Robot grasp synthesis algorithms: A survey," *International Journal on Robotic Research*, vol. 15, no. 3, pp. 230–266, 1996.

[12] A. Morales, "Learning to predict grasp reliability with a multifinger robot hand by using visual features," Ph.D. dissertation, Univertitat Jaume I, Castellón, Spain, 2004.

[13] M. T. Ciocarlie and P. K. Allen, "Hand posture subspaces for dexterous robotic grasping," *The International Journal of Robotics Research*, vol. 28, no. 7, pp. 851–867, 2009.

[14] C. Borst, M. Fischer, and G. Hirzinger, "Grasping the dice by dicing the grasp," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, pp. 3692–3697.

[15] S. El-Khoury and A. Sahbani, "Handling objects by their handles," in *Proceedings of IROS 2008 Workshop on Grasp and Task Learning by Imitation*, 2008.

[16] R. Pelossof, A. Miller, P. Allen, and T. Jebara, "An svm learning approach to robotic grasping," in *Proceedings of the IEEE Conference on Robotics and Automation*, 2004.

[17] N. Curtis and J. Xiao, "Efficient and effective grasping of novel objects through learning and adapting a knowledge base," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008.

[18] L. Bodenhagen, D. Kraft, M. Popović, E. Baśeski, P. E. Hotz, and N. Krüger, "Learning to grasp unknown objects based on 3d edge information," in *Proceedings of the 8th IEEE international conference on Computational intelligence in robotics and automation*, 2009.

[19] C. Dune, E. Marchand, C. Collowet, and C. Leroux, "Active rough shape estimation of unknown objects," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008.

[20] D. Kraft, N. Pugeault, E. Başeski, M. Popović, D. Kragic, S. Kalkan, F. Wörgötter, and N. Krüger, "Birth of the object: Detection of objectness and extraction of object shape through object-action complexes," *International Journal of Humanoid Robotics*, vol. 5, no. 2, pp. 247–265, 2008.

[21] D. Rao, Q. V. Le, T. Phoka, M. Quigley, A. Sudsang, and A. Y. Ng, "Grasping novel objects with depth segmentation," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.

[22] E. Başeski, N. Pugeault, S. Kalkan, L. Bodenhagen, J. H. Piater, and N. Krüger, "Using Multi-Modal 3D Contours and Their Relations for Vision and Robotics," *Journal of Visual Communication and Image Representation*, vol. 21, no. 8, pp. 850–864, 2010.

[23] A. Morales, P. J. Sanz, A. P. del Pobil, and A. H. Fagg, "Vision-based three-finger grasp synthesis constrained by hand geometry," *Robotics and Autonomous Systems*, vol. 54, no. 6, pp. 496 – 512, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/B6V16-4JF97MG-1/2/415fb6d35796bfc6591a5b60b996e972

[24] A. T. Miller and A. T. Miller, "Graspit!: A versatile simulator for robotic grasping," *IEEE Robotics and Automation Magazine*, vol. 11, pp. 110–122, 2004.

[25] J. Ponce and B. Faverjon, "On computing three-finger force-closure grasps of polygonal objects," *Robotics and Automation, IEEE Transactions on*, vol. 11, no. 6, pp. 868 –881, Dec. 1995.

[26] J. Jorgensen, L. Ellekilde, and H. Petersen, "Robworksim - an open simulator for sensor based grasping," in *Proceedings of Joint 41st*

International Symposium on Robotics (ISR 2010) and the 6th German Conference on Robotics (ROBOTIK 2010), Munich, 2010.

[27] L. Ellekilde and J. Jorgensen, "Usage and verification of grasp simulation for industrial automation," in *Proceedings of 42st International Symposium on Robotics (ISR)*, Chicago, 2011.