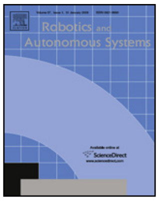
Contents lists available at SciVerse ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot



Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions

Luis Montesano[a,*], Manuel Lopes[b]

[a] I3A, Universidad de Zaragoza, Spain

[b] INRIA, Bordeaux Sud-Ouest, France

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Active Learning
Learning through experience
Grasping points

ABSTRACT

One of the basic skills for a robot autonomous grasping is to select the appropriate grasping point for an object. Several recent works have shown that it is possible to learn grasping points from different types of features extracted from a single image or from more complex 3D reconstructions. In the context of learning through experience, this is very convenient, since it does not require a full reconstruction of the object and implicitly incorporates kinematic constraints as the hand morphology. These learning strategies usually require a large set of labeled examples which can be expensive to obtain. In this paper, we address the problem of actively learning good grasping points to reduce the number of examples needed by the robot. The proposed algorithm computes the probability of successfully grasping an object at a given location represented by a feature vector. By autonomously exploring different feature values on different objects, the systems learn where to grasp each of the objects. The algorithm combines beta–binomial distributions and a non-parametric kernel approach to provide the full distribution for the probability of grasping. This information allows to perform an active exploration that efficiently learns good grasping points even among different objects. We tested our algorithm using a real humanoid robot that acquired the examples by experimenting directly on the objects and, therefore, it deals better with complex (anthropomorphic) hand–object interactions whose results are difficult to model, or predict. The results show a smooth generalization even in the presence of very few data as is often the case in learning through experience.

1. Introduction

Grasping is a core skill to develop robotic applications in flexible and dynamic manufacturing environments or in domestic applications such as service robotics. The ability to grasp objects involves several skills including motor control (planning, reaching) and perception (grasping point selection, object reconstruction, visual feedback). Due to the enormous variety of objects and grasping strategies, grasping is still an open problem. In the past years, standard techniques such as the one based on stability [1–5] have been combined and complemented with learning and adaptive strategies. In parallel, several insights from neuroscience and human development have been incorporated to computational models of infant grasp learning [6] and, more generally, on how humans deal with grasping [7]. There is strong evidence suggesting that humans easily determine, from their prior experience, what is the better way of grasping and the appropriate grasping point – i.e. the relative location of the hand and the object and the way to close the fingers around it – and can perform a grasp without any visual control, but using only visual planning [8].

Within this context, a current common approach to the grasping problem is to decouple, at least partially, the perception and control problems. First, the robot decides where to grasp the object based on the object perceived properties (e.g. visual descriptors, 3D models). Next, a combined reaching and grasping strategy tries to grasp the object at the previously selected point. This scheme has been widely used in many of the latest developed systems such as generic visual descriptors combined with planning [9], 3D object centered representations and reinforcement learning[10], 3D multi-modal features combined with a set of predefined grasp strategies [11], single image visual descriptors and visual servoing [12]. The independent treatment of both problems allows to reduce the computational complexity of the planning and learning algorithms.

This paper focuses on the perception step of the general grasping architecture described above and addresses the problem of where to grasp an object by actively learning visual descriptors for good grasping points. Learning grasping points has been approached mainly in two different ways. As a supervised learning

[*] Corresponding author.
E-mail addresses: montesano@unizar.es (L. Montesano), manuel.lopes@inria.fr (M. Lopes).

0921-8890/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2011.07.013

Please cite this article in press as: L. Montesano, M. Lopes, Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions, Robotics and Autonomous Systems (2011), doi:10.1016/j.robot.2011.07.013

problem that uses examples labeled by a user hand or experienced by the robot itself. The latter approach, based on autonomously acquired examples, is well suited for hands which cannot be easily simulated or when the grasping result cannot be easily predicted by an expert. The second approach is to use a reinforcement learning algorithm to optimize the grasp which, in turn, implicitly creates representations of good grasping points. A common requirement of both approaches is the need for a large number of training examples, which is a costly and time consuming process.

The paper proposes a method to actively select training examples based on the knowledge accumulated by the robot to efficiently learn grasping points. The main contributions of this paper are: (a) a new algorithm to estimate the full posterior of the grasping probability based on a set of (visual) features extracted from the object and (b) an active exploration strategy that reduces the amount of data required to learn where to grasp a set of objects. On one hand, the algorithm combines a Bayesian model of the probability of a grasp based on beta–binomial distributions with a non-parametric kernel based approach [13]. Based on a set of examples, it estimates the grasp probability for all potential grasping points. Its non-parametric approach makes local approximations of the underlying grasping probability and is able to capture highly non-linear dependences on the visual input features. On the other hand, the algorithm estimates the posterior predictive distribution instead of just the grasp probability. An active learning strategy exploits this extra information, such as the variance (or other measurements of information), to plan new grasps. Such a strategy allows a great reduction in learning time/samples necessary to converge to a good approximation of the function.

We illustrate the performance of our method in two different setups. First, we used a simulated one dimensional example to evaluate the method with a known ground truth. Next, we evaluated its performance on a grasping application using visual features extracted from a single image. We applied the algorithm to learning grasping points using Baltazar, a humanoid torso with an anthropomorphic hand. We recorded a set of trials where the robot tried to grasp different objects at several points. Based on this training set, the robot successfully predicted grasping points for new objects. On the other hand, the active strategy recovered the batch grasping probabilities with less examples. To provide a more systematic evaluation of our active learning approach, we also used the *Stanford synthetic object grasping point data* database [9].

The remainder of this paper is organized as follows. After discussing related work in Sections 2 and 3 presents our approach to grasp and the non parametric grasping point learning algorithm. Active learning and grasp point selection are described in Section 4. Section 5 describes the experimental setup with the real robot and Section 6 presents the experimental results on simulated data and on the real robot. Finally, in Section 7 we draw our conclusions and comment on future developments.

2. Related work

Early works on robot grasping used the geometry of the problem to model a stable grasp and control the robot to perform it [1–5]. After this initial research, the focus shifted to the relation between perception and action and several approaches tried to extract visual information to plan grasping actions: based on 2D information [14,15], approximating to known geometric shapes [16] or using range information [17]. Other works considered learning approaches to control the manipulator to the desired grasping position based on visual servoing techniques [18] or imitation [19]. Some approaches tried to learn a mixture of perception and action. For instance, [20] learns to grasp simplified superquadric shape approximations of objects. The work in [21] learns an object classifier based on the 2D appearance and the camera–robot calibration using Gaussian basis functions. Learning how to approach and grasp an object using reinforcement learning was used by Baier-Lowenstein [22].

The work in [23] implicitly used the concept of grasping points to define a reward function based on simple geometrical features of the object. Based on this reward, two learning algorithms applied at different spatial scales learned the appropriate controller to grasp the object using reinforcement learning on a PUMA robot and simple planar objects. As a by-product, the estimated Q-function, approximated using B-splines, encodes the utility of a particular grasping point. However, the reward function was based on simple heuristics and rules.

Instead of relying on very precise geometric features, the work in [9] proposed to learn correct grasping points directly from visual features using supervised learning techniques. In that work, the grasping point for a precision grip is computed as a function of the object's features extracted from a single image. The method uses a logistic regression to estimate the probability of a successful grasp at each potential image location. In order to recover the 3D location of the point, this operation is repeated from different points of view and the maximum likelihood grasping point is selected. The method has been extended to work with partial views of the object and range sensors [24].

Recently, other authors have exploited more sophisticated object representations such as 3D object reconstructions. In [10], objects and scenes are represented using early cognitive vision descriptors. For an object, the *likelihood* of grasping it is approximated over the whole object surface using a Gaussian process which also includes information about the gripper orientation. In [11], geometric constraints based on 3D features are combined with the appearance to hard wire an initial grasp which is then refined using robotic attempts to grasp the object.

In addition to previous works, the few other methods that address similar problems use only simulation. It is worth to mention [6], which presents a computational model on infant grasp learning. The method uses a Hebbian rule to learn the best grasping point and hand orientation in a simulated environment without considering the perceptual aspects of the problem. In [25], objects are described through rough descriptors obtained from the object bounding box. In simulation, the robot exhaustively tries all the grasping points (approach vectors) and keeps the good ones in a database. Unsupervised clustering of the objects provides a compressed database which is used to categorize new objects. The actual grasping is based on a 3D reconstruction of the bounding box and is adjusted for new objects. Some methods bootstrap the grasping learning problem using imitation or program by demonstration [26]. The demonstration provides a kinematic reconstruction of the hand during a grasp for a specific object. A map between different human–robot hand morphologies is used to allow replication of the same trajectory by the robot. Again, an exhaustive simulated search optimizes the reaching direction to apply the desired grasp on an object, thus learning a controller that maximizes the probability of grasping.

Since all the previous methods need a substantial amount of data, there has already been some research on active strategies to take advantage of the active nature of a robot learning through experience. In [27], active learning has been implemented using heuristics based on geometric information. The grasping probability was estimated with a nearest-neighbors approach on the memorized experiences. The exploration strategy reduces the difference between grasping classes probabilities. It suffers from lack of exploration and may be trapped in zones, where the real grasping probability is around 0.5. Recently, a two level architecture combined active learning and reinforcement learning to learn the grasping points and the corresponding policy for a single object [28] based on the object-centered representation in [10]. Their active strategy is based on the bandit problem and, similarly to our approach, they used a kernel based representation, namely Gaussian processes, to approximate the reward of a set of features and use them to select new samples.

This paper extends the beta–binomial regression algorithm of [13], shows its relation to kernel regression and develops active learning strategies for such a model. It shares ideas with the works of [28,9]. Similarly with [9], we learn a grasping probability and work on the feature space. This allows the use of negative information, and further improve the sample efficiency, and to generalize among objects in a transparent way. Our method allows a better fit of the distribution and has an active learning approach. In [28], the authors also present an active learning approach but, as their grasping densities are learned for single objects, the active exploration is based on a single object exploration and the generalization among objects needs to be explicitly addressed.

From an active learning for supervised regression point of view, the proposed model addresses the problem of constrained regression (since the probability should be within 0 and 1). The most standard way to solve such a constrained regression problem is to use logistic regression [29,30] which uses logits of probabilities. Active strategies for this type of techniques have been studied in [31] and pointed out a trade off between computational cost for experimental design based techniques and robustness issues that appear in more heuristic criteria. A support vector machine version of the logistic regression, the kernel logistic regression [32,33] uses the log-likelihood of the binomial distribution as the loss function and provides directly estimates of the probability. In this case, the typical active approach uses margins as the main criterion and selects input points that maximize them [34,35]. Active regression has also been used in linear regression, e.g [36]. By computing directly posterior distributions, the proposed method provides a natural interpretation of the confidence of the regressed function at each input point. Similar to Gaussian processes (see [37] and references therein), the model allows to devise active strategies based on the moments of the distribution, for example, the mean and the corresponding covariance.

3. Grasp point learning

This section presents an algorithm to estimate the grasp probability based on a set of examples. We assume that the robot is able to control its hand to a selected position on the image and then perform a reflex finger closure to grab the object. Thus, along the paper, a grasping point represents the location of the hand where the grasp action will take place for a fixed grasping strategy. The actual definition of a grasping point depends on the robot morphology. For instance, the grasping point description for a simple two finger gripper is simply the contact point. When dealing with a more complex anthropomorphic hand, the grasping point can be characterized as the position of a certain part of the hand or the wrist with respect to the object.

Although it is not required by the algorithm, we will also assume that the training data is autonomously acquired by the robot as follows. Given an object, the robot randomly (or actively, see Section 4) selects a potential grasping point (an image pixel), moves the hand to it and attempts to grasp the object. By repeating the previous procedure, the robot collects a set of n examples. We will refer to this set as the training dataset. Each example contains the grasping point, represented by a set of features $\mathbf{x} \in \mathcal{X}$, extracted from the image, and the grasp result. The objective is to use this training data to predict the result of grasping for a new input feature $\mathbf{x}$.

3.1. Beta–binomial model

For the time being, let us consider a single input point described by the feature vector $\mathbf{x}_i$. From a total of m trials, there are S_i positive results and U_i negative results, with $m = S_i + U_i$. Each of these trials is a Bernoulli experiment. Let p_i be the probability of a successful trial. Then, $p(S_i \mid p_i, m)$ is given by a binomial distribution of parameters p_i and m. Our objective is to estimate the posterior distribution of p_i given the examples, $p(p_i \mid \mathbf{x}_i, S_i, U_i)$.

The standard Bayesian approach to compute the posterior uses the beta distribution,

$$\text{Be}\,(p; \alpha, \beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)}, \quad p \in [0, 1] \tag{1}$$

as a prior, where $B(\alpha, \beta)$ is the beta function. α and β are shape parameters that can be interpreted as the number of success and failed trials encoding our prior knowledge. The binomial and beta distributions form a conjugate pair [38] and, therefore, the posterior is also a beta distribution with updated parameters $\alpha + S_i$ and $\beta + U_i$. Summarizing, the Bayes update gives the posterior

$$\text{Be}\,(p; \alpha + S_i, \beta + U_i) \propto \text{Bin}(S_i; p, S_i + U_i)\text{Be}\,(p; \alpha, \beta)\,. \tag{2}$$

3.2. Non-parametric smoothed beta distributions

We are now ready to introduce our algorithm to predict the probability of success using samples taken at different places of the input space $\mathcal{X}$. We define the input space as $\mathcal{X} \subseteq \mathbb{R}^d$ and denote input vectors as $\mathbf{x}$. The observations at a particular input vector $\mathbf{x}_i$ are the number of successful and failed trials, $\mathbf{y}_i = (S_i, U_i) \in \mathbb{N}_0^2$. We denote a set of input vectors as $\mathbf{X}_n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and the corresponding observations as $\mathbf{Y}_n = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$. The objective is to estimate the posterior $p(p_* \mid \mathbf{x}_*, \mathbf{X}_n, \mathbf{Y}_n)$ at an arbitrary point $\mathbf{x}_*$. Formally, we are looking for a map $f : \mathcal{X} \to \mathcal{B}_{\alpha\beta}$ where $\mathcal{B}_{\alpha\beta}$ is the space of beta distributions parametrized by α and β.

To predict the posterior probability at point $\mathbf{x}_*$, we apply the Bayes rule,

$$\begin{aligned} p(p_* \mid \mathbf{x}_*, \mathbf{X}_n, \mathbf{Y}_n) &\propto p(\mathbf{Y}_n \mid p_*, \mathbf{x}_*, \mathbf{X}_n)p(p_* \mid \mathbf{x}_*, \mathbf{X}_n) \\ &\approx \prod_{i=0}^{n} p(\mathbf{y}_i \mid p_*, \mathbf{x}_*, \mathbf{x}_i)p(p_*), \end{aligned} \tag{3}$$

where the approximation of the last expression assumes that, given the parameter p_* at $\mathbf{x}_*$, the results observed are independent of each other. This simplification of the model is required to derive the closed-form expression based on the beta–binomial presented below. The prior distribution $p(p_* \mid \mathbf{x}_*, \mathbf{X}_n)$ is just a beta with parameters α_0 and β_0 independent of the training data. For instance, if we set both parameters to one, the prior becomes a uniform distribution.

The likelihood term $p(\mathbf{y}_i \mid p_*, \mathbf{x}_*, \mathbf{x}_i)$ represents the likelihood of the observations $\mathbf{y}_i$, given the input features $\mathbf{x}_i$ and $\mathbf{x}_*$ and the success rate p_*. To model this dependency, we have to add some structure to the model, in particular, we assume that the probability is smooth, given the feature space. Based on this extra structure, we take a kernel based the non parametric approach [39] to extrapolate the parameter p_* of the binomial distribution from the features of the training dataset to new ones. We define a kernel based diffusion of the sufficient statistics ($\alpha_{\mathbf{x}_i}$ and $\beta_{\mathbf{x}_i}$) of the beta posterior of an input point $\mathbf{x}_i$ in the database to a new input point $\mathbf{x}_*$.

$$\alpha_{\mathbf{x}_*} = K(\mathbf{x}_i, \mathbf{x}_*)\alpha_{\mathbf{x}_i}, \qquad \beta_{\mathbf{x}_*} = K(\mathbf{x}_i, \mathbf{x}_*)\beta_{\mathbf{x}_i} \tag{4}$$

where $K : \mathcal{X} \times \mathcal{X} \Rightarrow [0, 1]$. Intuitively, when the kernel is a decreasing function of the distance between $\mathbf{x}_1$ and $\mathbf{x}_2$, the diffusion process keeps the same beta parameters for the same point and decreases as the point is further in the feature space. As a result, when all the kernels vanish, the prediction converges to the prior $\text{Be}(\alpha_0, \beta_0)$. The likelihood function of each individual observation

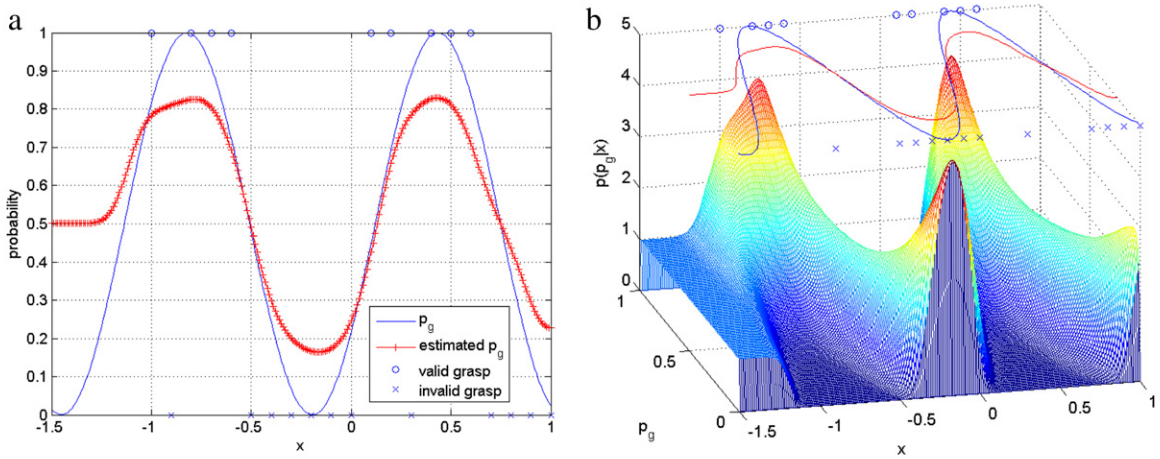


Fig. 1. Approximating a sinus varying p in a one dimensional input space. (a) Estimated mean. The 0–1 blue points are the observations generated from a Bernoulli experiment, using the true p (blue line). Failures are represented by crosses and successes by circles. The red line with marks is the approximated mean computed from the posterior. (b) Predicted posterior beta distributions for each point along x.

is modeled as

$$p(\mathbf{y}_i \mid p_*, \mathbf{x}_*, \mathbf{x}_i) = \mathrm{Bin}(S_{*i}; p_*, S_{*i} + U_{*i}) \tag{5}$$

where the variables $S_{*i} = K(\mathbf{x}_*, \mathbf{x}_i)S_i$ and $U_{*i} = K(\mathbf{x}_*, \mathbf{x}_i)U_i$ are the number of successful and failed trials at point $\mathbf{x}_*$, given the examples observed at $\mathbf{x}_i$.

Finally, the predicted posterior probability at point $\mathbf{x}_*$ can be computed as

$$p(p_* \mid \mathbf{x}_*, \mathbf{X}_n, \mathbf{Y}_n)$$

$$\propto \prod_{i=0}^{n} \mathrm{Bin}(S_{*i}; p_*, S_{*i} + U_{*i})\mathrm{Be}(p_*; \alpha_0, \beta_0) \tag{6}$$

$$= \mathrm{Be}\left(p_*; \sum_{i=1}^{n} S_{*i} + \alpha_0, \sum_{i=1}^{n} U_{*i} + \beta_0\right). \tag{7}$$

Fig. 1 shows an example of the algorithm predictions for a one dimensional feature space. The p_i is a sinusoidal function of the input x. The 1–0 observations are generated through a binomial distribution based on the true p_i at that point. Fig. 1(a) shows the true function and the predicted mean. An important feature of our algorithm is that it estimates the full posterior distribution at each feature (see Fig. 1(b)). This is important because it provides an associated notion of confidence, or uncertainty, about the predicted probability. The figure also illustrates how the algorithm tends to the prior distribution in the absence of measurements. In the figure, this occurs for $x < -1$, where the predicted mean value of 0.5 is due to the uniform prior.

When the number of trials per training point is one, it is interesting to re-write our approximation of the mean using the sufficient statistics of the beta distribution of Eq. (7),

$$\bar{p}_* = \frac{\sum_{i=1}^{n} S_i K(\mathbf{x}_*, \mathbf{x}_i) + \alpha_0}{\sum_{i=1}^{n} K(\mathbf{x}_*, \mathbf{x}_i) + \alpha_0 + \beta_0}, \tag{8}$$

which has been simplified using the fact that $S_i + U_i$ is one for every point. The expression illustrates that the estimated mean is basically equivalent to a kernel regression of the probability for each input point. However, our method keeps two individual accumulators to estimate virtual trials over the whole input space. This extra information, together with the interpretation of Eqs. (7), provides a beta distribution to model the probability at every input point.

On the other hand, it is interesting to note that the extension to the multi-class case is direct. One simply needs to accumulate events for each class independently to compute the proportions. For an M class problem, the probabilities estimated are

$$\bar{p}_{j*} = \frac{\sum_{i=1}^{n} S_i^j K(\mathbf{x}_*, \mathbf{x}_i) + \alpha_j}{\sum_{i=1}^{n}\sum_{j=1}^{M} S_i^j K(\mathbf{x}_*, \mathbf{x}_i) + \sum_{j=1}^{M} \alpha_j}, \quad j \in 1, \ldots, M-1 \tag{9}$$

where S_i^j is the number of cases for class j at input point $\mathbf{x}_i$. The virtual counts define at each point a Dirichlet distribution with parameters analogous to the binary case, whose mean corresponds to Eq. (9). The probability of the last class is simply $p_{M*} = 1 - \sum_{j=1}^{M-1} p_{j*}$. The computational and memory cost increases with the number of classes, since it is necessary to approximate the counts for every class independently.

3.3. *Kernel parameter selection*

Besides the beta distribution encoding our prior knowledge about the problem, the algorithm requires to set the parameters θ that define the kernel $K(\cdot, \cdot)$. We estimate these parameters from the training data using a cross-validation approach. For each point, we compute the posterior distribution $p(p_i \mid \mathbf{x}_i, \mathbf{X}_{n,-i}, \mathbf{Y}_{n,-i})$, where $\mathbf{X}_{n,-i}$, $\mathbf{Y}_{n,-i}$ represent the input features and observations of the dataset except $\mathbf{x}_i$ and $\mathbf{y}_i$ respectively. Then, we use a minimum least square criterion[1] between the predicted mean $\bar{p}$ at each point and the empirical one observed $\hat{p} = \frac{S_i}{S_i+U_i}$,

$$L_\theta(X_n, Y_n) = \sum_{i=1}^{n} (\hat{p}_i - \bar{p}_i)^2 = \sum_i \left[\frac{\alpha_i(\theta)}{\alpha_i(\theta) + \beta_i(\theta)} - \hat{p}_i\right]^2 \tag{10}$$

with $\alpha_i(\theta) = \sum_{j \neq i} k_{ij}(\theta)S_j + \alpha_0$, $\beta_i(\theta) = \sum_{j \neq i} k_{ij}(\theta)U_j + \beta_0$ and $\hat{p}_i$ is the empirical approximation of parameter p_y given by $\frac{S_i}{S_i+U_i}$.

In our experiments, we used a Gaussian kernel with a diagonal covariance matrix, that is, with independent bandwidths for each dimension of the features. The function was minimized using the *active set* method [40] and positive constraints for the kernel bandwidths. Other constrained optimization methods such as the *interior point* provide similar results.

[1] We also minimized the pseudo-likelihood cross validation function, but the results were worse than using the least square one.

Please cite this article in press as: L. Montesano, M. Lopes, Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions, Robotics and Autonomous Systems (2011), doi:10.1016/j.robot.2011.07.013

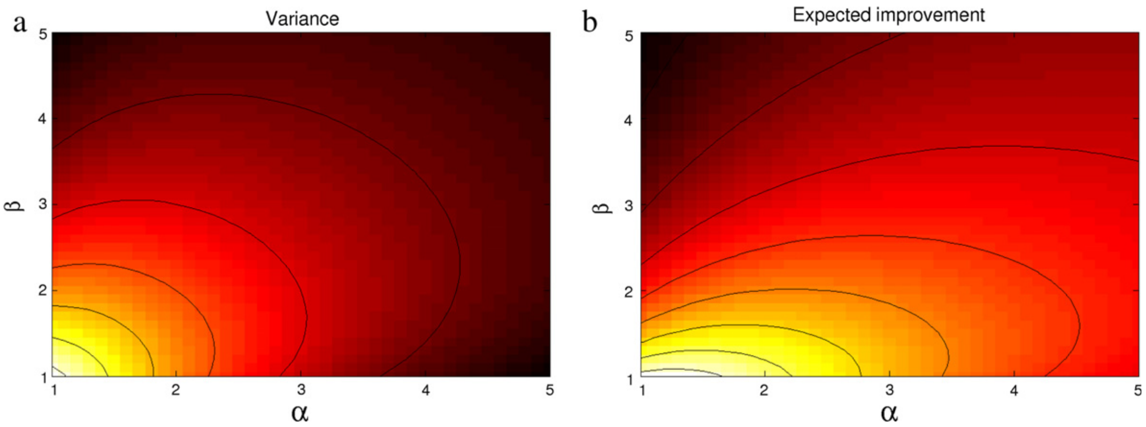


Fig. 2. Value of an input point according to its predicted α and β. Darker points have a smaller value for learning. (a) Variance. (b) Expected improvement.

Table 1
Different active learning criteria for Be(p; α, β).

Criterion	$I(\cdot)$
Random	*cte*
Entropy	$\log B(\alpha,\beta)-(\alpha-1)\psi(\alpha)-(\beta-1)\psi(\beta)+(\alpha+\beta-2)\psi(\alpha+\beta)$
Variance	$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$
Expected improvement	$f(\bar{p}_i, Var(p_i))$

4. Active learning

This section addresses the problem of actively selecting the input points based on the current knowledge. We will focus on the pool based case where there is a set of unlabeled points $\mathbf{X}_o=\{\mathbf{x}_{o1},\ldots,\mathbf{x}_{om}\}$ to choose from. The general active learning selection equation we will use is

$$\mathbf{x}_n=\arg\max_{\mathbf{x}_i\in\mathbf{X}_o} I(\mathbf{x}_i), \tag{11}$$

where $I(\mathbf{x})$ is a measure of the *improvement* in the regression after trying the point $\mathbf{x}$.

We will consider two different cases. In the first one, the objective is to improve the approximation of the function over all the input space. The second one considers the situation where some parts of the space are more important than the others. For instance, consider the case where a robot learns to grasp objects and, therefore, its main interest is to discover areas with a high probability of a successful grasp. In this situation, it is sensible to focus the exploration first on those parts of the input space which are, according to the previous data, more likely to have a high probability. This is in essence, an exploration-versus-exploitation problem where we want to explore as much as possible by converging to the true distribution but, due to the lack of resources, also give priority to those parts of the feature space which may have a high probability rate (or any other criteria). Table 1 summarizes some of the most common criteria and the corresponding expressions for a beta distribution.

Random selection allows to converge, in the infinity, to the desired distribution but it can be very inefficient. With our learning method, we have not only access to the grasp probability but also the confidence on its estimate. Using such information, it is possible to make more informed exploration. Fig. 2(a) shows the variance of a beta distribution with different parameters α and β. In this case, a low variance indicates that we are certain about the parameter p even if the outcome of the action will be uncertain. In a nutshell, the variance strategy selects features with the lowest sum $\alpha+\beta$. On the other hand, the entropy of a beta distribution is based on the *digamma function* $\psi(\cdot)$. Entropy has a similar behavior as the variance, but for a fixed number of trials favors those parts of the space with equal α and β, i.e. with a probability close to 0.5.

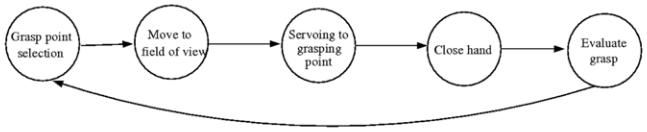


Fig. 3. Experimental protocol used during the experiments.

An alternative approach is to select points that allow you to quickly discover those regions that match a particular criterion e.g., high probability of success. This can be achieved using the idea of *Expected Improvement* (EI) suggested by Mockus [41] in its original form and with other flavors by Oudeyer et al. and Kapoor et al. [42,43]. In our case, we want to favor the exploration of points that can provide a high probability and, at the same time, take into account the uncertainty provided by the predictions. We propose the following criteria

$$I(p_i)=\bar{p}_i Var(p_i) \tag{12}$$

$$I(p_i)=|\bar{p}_i-0.5|Var(p_i). \tag{13}$$

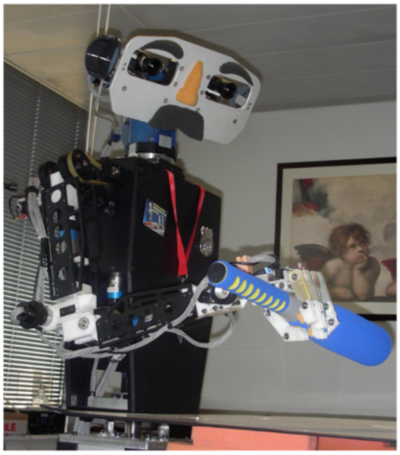
The previous expressions combine two components to favor the exploration of those areas where the probability is close to what we expect, for instance, a high probability of success or where the uncertainty is big (Eq. (12)) or when is close to the extreme values (Eq. (13)). Fig. 2(b) shows the value function for Eq. (12). As a result, the exploration proceeds by first selecting points with probability slightly over the prior with a small number of samples, then points with a small number of samples and finally all the others. Note that this method is still able to explore all the (available) feature space. A point with high probability is selected only a number of times before its uncertainty is reduced to a level that favors exploration to points with little evidence.

5. Experimental setup

5.1. Robot

We used Baltazar, a humanoid torso composed of a binocular head and an arm-hand [44]. The anthropomorphic robot arm and hand have 6° and 4° of freedom, respectively. In the experiments, the robot followed the protocol depicted in Fig. 3. It consists of five different steps. First, the robot selects a grasping point and moves the hand to the field of view using an open loop control. Then, it switches to a closed-loop control to position the hand to the selected point in the image. When the object is reached, it performs a preprogrammed closure phase of the hand fingers. Finally, the

Please cite this article in press as: L. Montesano, M. Lopes, Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions, Robotics and Autonomous Systems (2011), doi:10.1016/j.robot.2011.07.013



type	parameter	number
Gaussian	iso. diag. cov.	8 scales
Gaussian top	diag. cov.	8 scales
Gaussian bottom	diag. cov.	8 scales
Gaussian left	diag. cov.	8 scales
Gaussian right	diag. cov.	8 scales
Gaussian Laplacian	diag. cov.	8 scales
Gaussian	cov. *diag*([*d sd*])	8 scales×6 skew
Gaussian Laplacian	cov. *diag*([*d sd*])	8 scales×6 skew
Sobel	-	5 orientations
Laplacian	-	2 scales
Total	151	

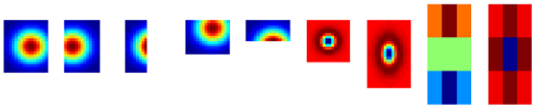


Fig. 4. Experimental setup. The left image shows Baltazar, the humanoid torso robot used in the experiments. The right table shows a summary of the filters which are shown in the bottom right figure.

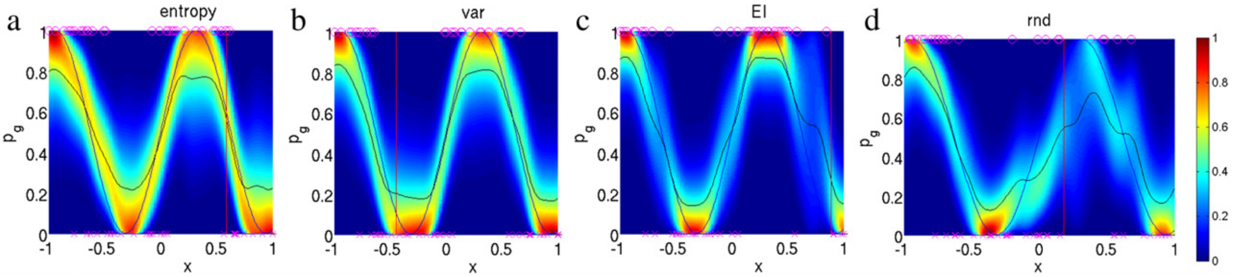


Fig. 5. The posterior over the parameter p after 60 queries. Each column provides the posterior for the corresponding input point x. The points were selected based on (a) entropy, (b) variance, (c) the EI and (d) randomly.

robot raises the arm and the grasp is evaluated. If the object does not fall during the motion, the grasp is considered a success and a failure otherwise. Fig. 4 shows Baltazar performing the last phase of the protocol after grasping a bat.

Based on the previous protocol, we implemented a controller to reach the object from the top until contact is made, which triggers the hand closure phase. Contact detection is necessary due to the depth ambiguity in an image. The hand orientation was fixed and did not consider the object. Finally, to ease the tracking of the robot hand, we used the *ARToolKit* tracker [45]. We place a marker on the robot hand that actually defines the correct grasping point in terms of the relative position of this hand point and the object. With calibrated cameras, the *ARToolkit* tracker provides 3D information, but we use only the image position.

5.2. *Computer vision features*

Each pixel in the image is a potential grasping point described by a set of features. The main constraint required by our algorithm is that, these features must capture enough object information so as to make p smooth. In our case, the selected features are generic filters applied directly to the saturation channel of the image. In this way, features are general, do not include any a priori object knowledge and are robust to different illumination conditions. In addition to this, they are robust, fast and easy to extract. We use the bank of 151 filters shown in Fig. 4. It is worth to point out that results are highly dependent on the input features. For the evaluation of active strategies, we have selected single image features that do not require to have multiple viewpoints of the object or an a priori learned model. Although they contain less information than more sophisticated ones, they served to evaluate the advantages of the method to directly transfer information between different objects and, as the experimental results suggest, to actively learn grasp points with less examples.

6. Experimental results

This section evaluates the proposed regression method and the active exploration strategies. We present two different evaluations. The first one is a simple 1 dimensional case that shows how the different criteria proposed in Section 4 explore the input space. The second case is an application to grasping point learning, where a robot learns from experimentation which are the correct points to grasp different objects. We provide results using a synthetic dataset and real data obtained with a humanoid robot. This example works on a 151 dimensional feature space and shows that our algorithm can deal with such high-dimensionality. We study the ability of the algorithm to learn the grasp probability and the gain of an active approach to discover good grasping points compared to random exploration.

6.1. *One dimensional simulation example*

This section provides some experimental evidence of the behavior of the different criteria of Table 1 on a simple mono-dimensional simulated example. We consider that the parameter varies according to $p = (1 + \sin(cx))/2$ where c is a constant. The input space is $\mathcal{X} = [-1, 1]$.

Fig. 5 shows the posterior of p after 60 queries. The random exploration provides worse results than any of the active strategies. As expected, the entropy focus more on the uncertain outcome regions and, consequently, it has more peaked posterior

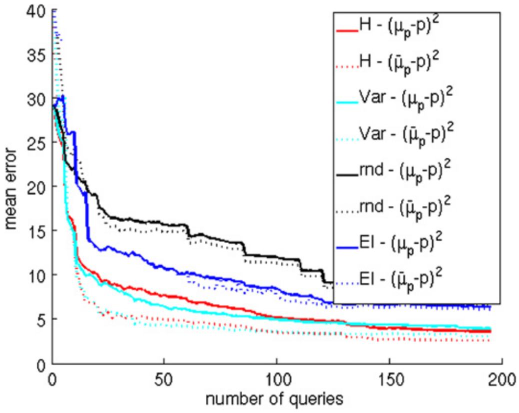


Fig. 6. Sum of squared errors over the entire input space for the mean (μ) and mode $(\tilde{\mu})$ of the parameter p. The figure shows the average over 20 runs for the different exploration methods after 200 queries.

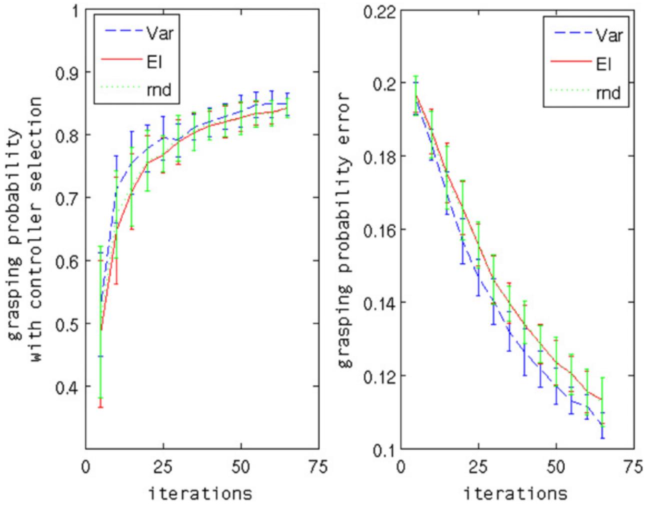


Fig. 7. Evolution of the learning of the grasp probability (right) including controller parameters (left).

on the areas around 0.5. The EI criteria used in this figure is $EI(x) = |\hat{p}_x - 0.5|var(p_x)$ which favors exploration when p is far from 0.5 and has high variance. The queries, therefore, concentrate along the peaks. Finally, the variance has a more homogeneous precision along the whole space.

Fig. 6 shows the evolution of the error with the number of queries. The curves correspond to the average over 20 trials. The figure shows how the error reduces faster for any of the active approaches. The variance provides the best results at the beginning. The entropy criterion equals the variance with enough number of samples. It is important to note that the EI criterion is not designed for a good approximation of the function, but only of those areas where the exploration focus on. Fig. 5 shows that in the peaks of the function, the EI criteria is better than the variance at the expense of a worse approximation in the other regions.

6.2. Including free parameters for controller selection

We now evaluate the algorithm in a more complex simulated model where, in addition to the input features, there also exist other free parameters that can be optimized based on the input features. This case arises, for instance, when there are some parameters controlling the grasping behavior. For learning, the database contains examples of input features together with the free parameters. However, when exploiting the learned model, only the input features are fixed while the free parameters have to be selected based on the training examples. This is done by recovering a function of the free parameters conditioned on the input features and optimizing according to some criterion, e.g the probability of success,

$$\hat{\mathbf{x}}\hat{\rho} = \arg\max_{\mathbf{x},\rho} p(y_i|\mathbf{x},\rho). \qquad (14)$$

We now compare the active learning in a three-dimensional input feature $\mathbf{x} = (x_1, x_2, x_3)^T$ representing the object features and a one dimensional free parameter ρ corresponding to a continuous parameter describing the action. The probability of success depends on $\mathbf{x}$ and ρ. For a given $\mathbf{x}$, it attains its maximum value at $\text{atan}(x_1/x_2)$ and decreases following a Gaussian function. The other dimension is noise. Since we have a single parameter, each input feature $\mathbf{x}$ provides a one dimensional function of the free parameter ρ for the grasping probability similar to the ones depicted in Fig. 5. The active learning must, therefore, optimize over a set of functions one per input feature to obtain the best ρ. This is done by randomly sampling the free parameter.

The results are shown in Fig. 7, comparing the active learning of both the input features and the free parameter using a random strategy and an active learning one based on the variance. The figure shows the mean and variance over 10 different runs. The left figure shows the number of successful grasps using the learned model equation (14) as a function of epochs, where each epoch corresponds to 10 new training examples. The results show that in this case, the variance provides a better result than the EI, while the latter is almost identical to a random strategy. The right figure shows the prediction error of the grasping probabilities over the whole space. Again, the variance provides a better result than the EI which is close to random. We would like to remark that the active learning criteria do not exploit explicitly the control parameter and, therefore, we expect that the gain of active strategies will improve by selecting ρ using the predicted probabilities.

In addition to evaluating the active learning approach in a more complex setup, this section shows that the proposed method goes beyond simply selecting the grasping point. It can also incorporate information about other variables and can be used to select parameters of a controller. We note, however, that the direct extension of this approach to high-dimension controllers will not be very robust or efficient.

6.3. Learning of grasping probabilities

We now present the results of learning grasping points using the proposed model and study the benefits of an active point selection. The general scheme for this problem is the following. Objects are represented using the visual features of Section 5, thus reducing our problem to a regression problem on a 151 dimensional space. In order to speed up exploration, we bootstrap the learning with an initial labeled dataset. It contained four negative examples (all in the background) and a single positive one. They were equal for all the exploration strategies. From this initial dataset, only one new grasping point can be tried for each new image of an object available to the robot. The object is always placed in the same position with respect to the robot and, consequently, it is centered in the image with a non-cluttered background. For each image pixel, the corresponding unlabeled feature is computed to create a set of potential grasping points. The best point according to the selection criterion is tried by the robot and stored in the dataset with the corresponding result. Note that each time the set of unlabeled points is different due to the different location of the same object and illumination conditions.

We compare the criteria of Section 4, variance and EI strategies, with a Gaussian random sampling around the image center. For

Please cite this article in press as: L. Montesano, M. Lopes, Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions, Robotics and Autonomous Systems (2011), doi:10.1016/j.robot.2011.07.013

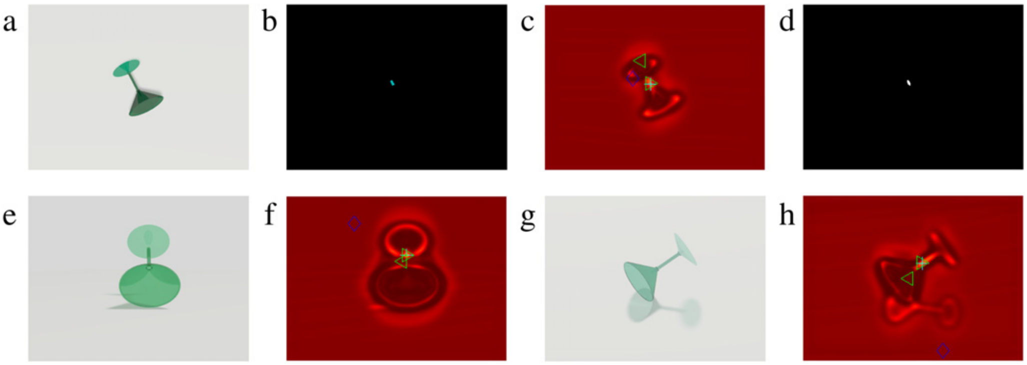


Fig. 8. Martini glass from the Stanford synthetic object grasping point data. Top row—(a) Martini glass, (b) true grasping points, (c) estimated grasping probability and (d) point with $\hat{p}_g > 0.5$. Bottom row shows examples from a different point-of-view. Cross symbol (+) shows the best, and correct, grasping point. The correct grasping points for views (a), (e) and (f) are always located in the central tube joining the top and the bottom of the glass. Learning was done using the EI strategy. The lighter the color, the highest the probability of a successful grasp.

Table 2
Grasping point predictions for the Martini glass.

	Graspable	Non-graspable
EI	16/18	2/3
var	12/18	2/3
rnd	1/18	3/3

a fair comparison, the random sampling exploits the fact that the object is always in the center of the image.[2]

6.3.1. Simulated data

We first tested the active strategies on the *Stanford synthetic object grasping point dataset* [9]. This simplifies the comparison among strategies, since with a real robot it is impossible to repeat exactly the same experiment and the ground truth is not available. The dataset is composed by a set of labeled images for simulated objects obtained from different points of view. The labels indicate if the image point is a grasping point, or not, for a gripper. Fig. 8(a) and (b) show a martini glass and its corresponding annotated image.

We now compare the results of the different exploration techniques. For each image, a sample was selected according to the selected strategy (random, variance or EI). Each sample is added to the training set and used for prediction at the next image. The procedure is repeated through the 21 images 24 times (epochs), which makes 504 actively selected points. By looping through different images, we also tested the algorithm and the exploration strategies ability to generalize between different points of view.

Table 2 shows that the EI found the correct grasping in 16 of the 18 images containing a graspable object. This higher ratio of success of the active exploration in relation to the one for variance and random shows that this approach is able to get more meaningful information with less amount of data. For images not containing graspable objects, results are similar among the EI and variance.

These results confirm the empirical evidence found on the one dimensional example. The EI is able to find the best grasping points faster and has a better approximation of the function for points graspable with higher probability. Although the price to pay is a worst approximation of the function for low probability grasping points, Fig. 8(c, f, h) show that the predicted good and bad grasping points actually match the ground truth (i.e. the central part of the glass as shown in Fig. 8(b)).

The same analysis was performed with other objects in the database. Similar results were obtained for the pencil, the stapler and the coffee mug. However, the model learned for the cereal bowl failed to learn positive examples. The main reason is the texture of the object (a spiral from the center to the outside) that produced very similar features all along the object and were not able to detect the edge of the bowl.[3]

6.3.2. Real robot

Finally, we provide results of active learning with data acquired directly with a robot and the objects in the training database (see Fig. 9). This robot has a more complex hand geometry that results in more complex hand–object relations. For each strategy, the robot was presented with the same object several times and tried to grasp it in the selected grasping point.

Fig. 10(a) shows the differences between the EI and random exploration strategy for a box after 20 actively explored grasps. Since no ground truth is available, we used the results of the batch learning as a baseline for the active approach. The EI strategy explored more of those areas with high grasping probability. As a result, these areas have been correctly identified and are similar to the ones in the batch solution. This is done at the price of having a poorer approximation in those areas with a low grasping probability. The random strategy on the other hand, detected better the low probability grasping points but its estimate for good grasping points is worse.

In addition to this, we show the *accumulated reward*, i.e. the number of successful grasps achieved during learning (see Fig. 10(b)). As mentioned before, the EI strategy implements a trade-off between exploration and exploitation. The higher number of successful grasps for the EI strategy reinforces the idea that the EI is correctly focusing on the part of the image which are, potentially, good grasping points. Similar results were obtained for the other objects in the training database where the EI criterion detected good grasping points faster than random exploration.

Finally, we present some results of the prediction capabilities of the model after a batch learning with a total of 550 trials over the objects in the training set of Fig. 9. First, we estimated the optimum kernel parameters for this dataset using an active set method and the cost function defined in Section 3.3. We initialized the kernel bandwidths with random values. The number of iterations until convergence was around one hundred. Very similar results were obtained from different initializations, in

[2] A pure random strategy produced worse results, since a lot of sampling occurred out of the object.

[3] The features were mainly designed to detect edges and separate the object from the background.

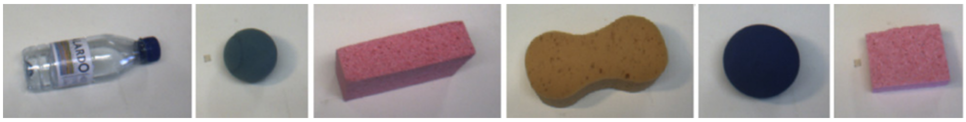


Fig. 9. Objects used to acquire the training dataset. (Images at different scales.)

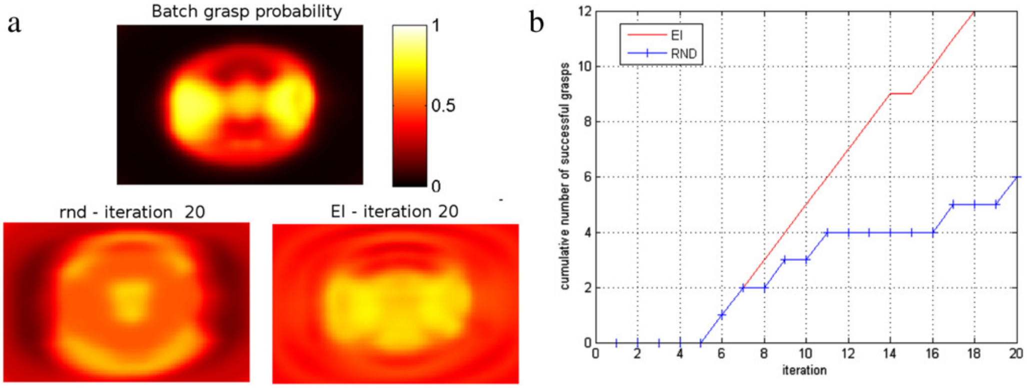


Fig. 10. Active learning for the box object in the training database. (a) (Top) Batch result used as ground truth. (Bottom) The grasping probabilities estimated after 20 samples using random exploration (left) and the EI (right). (b) Number of successful trials during active learning of the box using random and EI strategies.

Table 3
Grasping probability predicted by the algorithm and the empirical observation (results for 5 trials on test objects).

#	Object	Grasping success (%)	
		Predicted (%)	Observed (%)
1	Bat	82	80
2	Can	64	0
3	Salt shaker	82	0
4	Tea box 2	86	100
5	Pencil box	84	80
6	Plastic cup	80	80
7	Basket	50	80
8	Tea box 1	87	100
9	Ball	91	80

terms of the set of selected filters and the corresponding weights. With slight variations, the number of kernel bandwidths different from zero was 12 in average, thus substantially reducing from the initial 151.

Then, using our beta regression model, we experimented to grasp several new objects at the point with highest predicted probability. Fig. 11 shows the test objects and Table 3 summarizes the results by showing the prediction and the empirical grasping probabilities over 5 trials. The point predicted for objects 4, 5, 6, 8 and 9 was always located in a similar part of the object and was a good location to perform the grasp (see Fig. 12(a) for the predictions of two images of the same ball). The same happened with the bat (object 1) which is a much bigger object, but whose grasping point was correctly predicted where the hand best fits it (see Fig. 12(a) and (b)).

When objects are very different (in the feature space), the algorithm gives just the prior information. This can be seen on the basket case (object 7) where the posterior of the best grasping point being basically the prior (see Fig. 12(b)). However, the method was able to reject many negative points based on prior experience which allows the robot to grasp in a point which, in this particular case, allowed grasping the basket.

The prediction for objects 2 and 3 in Table 3 was wrong, since the robot could not grasp the objects at the selected points. For the salt shaker (Fig. 12(b)), the variance of the posterior was low which indicates that it is close to some objects in the training dataset. This indicates that the local used features were not able to reflect important information that affected the grasp: the rigidity of the material and the conic shape which make the object slide from the hand. This was true also for the can, but here the posterior was much more uncertain, reflecting the fact that the object is not very well represented in the database.

7. Conclusions

This paper introduces an algorithm to compute the posterior of the smoothly varying parameter (in a compact feature space) of a Bernoulli distribution from samples of binomial distributions taken at different places of the feature space. The method allows to model prior knowledge through the parameters α_0 and β_0 of a beta distribution. The rest of the parameters, and the kernel bandwidths are automatically selected based on the training data through the minimization of a least square loss function. Based on the predicted distributions, we also discussed and proposed active exploration strategies to efficiently explore the feature space. The method can be applied generally in any situation where we have a success/failure situation depending on a feature space. For instance, predicting if a given transmission will be successful or not depending on atmospheric conditions, or in which regions of a feature space a given image recognition algorithm works.

We evaluated the method in a learning by exploration robot scenario. The experimental results confirm those obtained in simple simulations. For the single image visual features used to described the object, they showed how active learning strategies greatly reduce the amount of data required for prediction. By favoring expected positive examples, the algorithm rapidly discovers good grasping points and provides an anytime learning algorithm for grasping. Since the quality of a grasping point depends on the features used, future work will study the benefits of the proposed active framework for more complex features such as 3D or global object descriptors. The proposed method can directly deal with motion controller parameters such as the hand orientation and allow for a fast tuning based on the object features.

Also, we are currently studying connections with multi-arm bandit algorithms for planning optimal exploration strategies as used in [28], but exploiting the beta–binomial model. This model has already been successfully applied to learn policies from demonstrations [46] and to learn user preferences [47]. We plan to explore the active strategy in these tasks where the positive reward is scarce and exploration is difficult without a good initialization.

Please cite this article in press as: L. Montesano, M. Lopes, Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions, Robotics and Autonomous Systems (2011), doi:10.1016/j.robot.2011.07.013

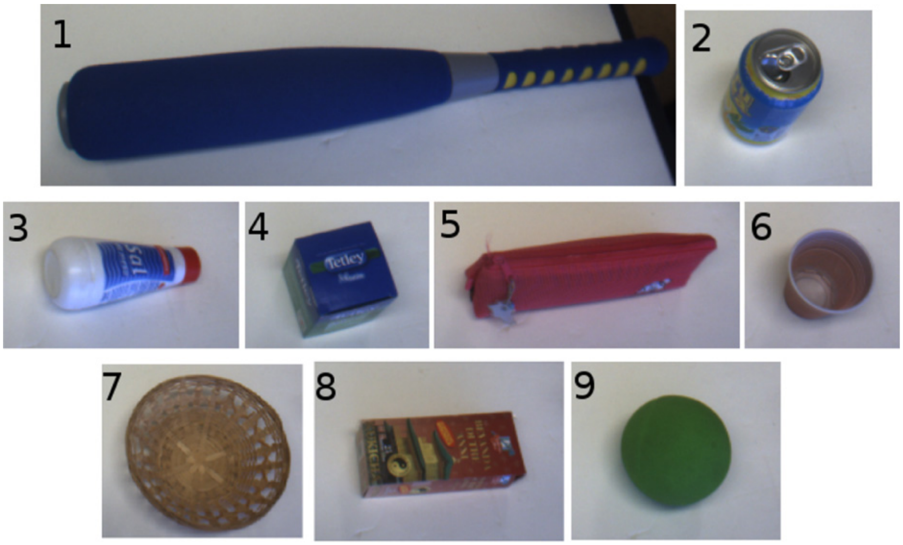


Fig. 11. Objects used to test the generalization capabilities of our learning system when trained with the objects from Fig. 9.

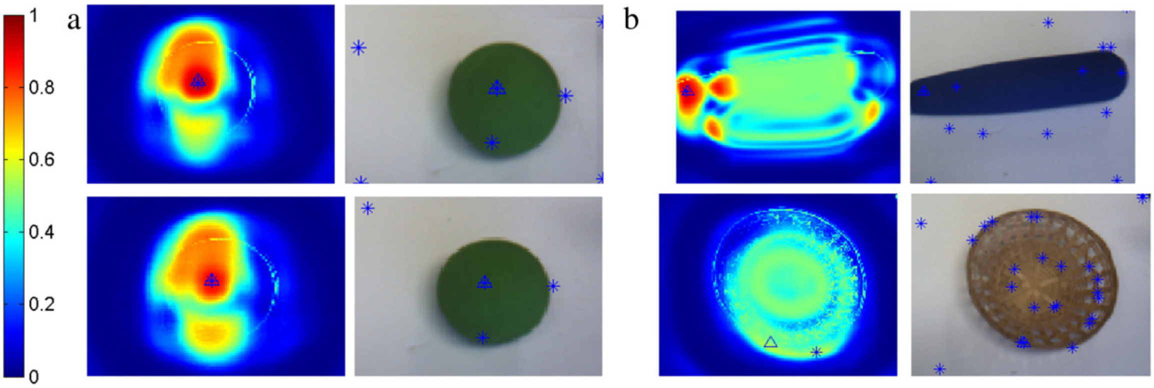


Fig. 12. Prediction for some of the test objects. The triangle and the asterisk indicate respectively the maximum predicted p and the best grasping point taken into account the control error. (a) Prediction for the ball at two different images. (b) Predictions for the bat and the basket.

The smoothing properties of the kernel are a way to transfer information from both positive and negative examples.

Acknowledgment

This work has been partly supported by the EU projects HANDLE and FIRST-MM.

References

[1] R.M. Murray, Z. Li, S.S. Sastry, A Mathematical Introduction to Robotic Manipulation, CRC, 1994.
[2] A. Bicchi, V. Kumar, Robotic grasping and contact: a review, Proc. IEEE International Conference on Robotics and Automation 1 (2000) 348–353.
[3] A. Bicchi, On the closure properties of robotics grasping, International Journal of Robotics Research IJRR 14 (4) (1995) 319–334.
[4] M. Buss, H. Hashimoto, J.B. Moore, Dextrous hand grasping force optimization, IEEE Transactions on Robotics and Automation 12 (3) (1996) 406–418.
[5] M. Cutkosky, On grasp choice, grasp models, and the design of hands for manufacturingtasks, IEEE Transactions on Robotics and Automation 5 (3) (1989) 269–279.
[6] E. Oztop, N.S. Bradley, M.A. Arbib, Infant grasp learning: a computational model, Expositions Brain Research 158 (2004) 480–503.
[7] E. Oztop, M. Kawato, M. Arbib, Mirror neurons and imitation: a computationally guided review, Neural Networks 19 (3) (2006) 254–271.
[8] M.E. McCarty, R.K. Clifton, D.H. Ashmead, P. Lee, N. Goubet, How infants use vision for grasping objects, Child Development 72 (4) (2001) 973–987.
[9] A. Saxena, J. Driemeyer, A.Y. Ng, Robotic grasping of novel objects using vision, International Journal of Robotics Research IJRR (2008).
[10] J. Piater, S. Jodogne, R. Detry, D. Kraft, N. Krüger, O. Kroemer, J. Peters, Learning visual representations for perception-action systems, International Journal of Robotics Research (2010).
[11] M. Popovic, D. Kraft, L. Bodenhagen, E. Baseski, N. Pugeault, D. Kragic, T. Asfour, N. Kruger, A strategy for grasping unknown objects based on co-planarity and colour information, Robotics and Autonomous Systems 58 (5) (2010) 551–565.
[12] L. Montesano, M. Lopes, A. Bernardino, J. Santos-Victor, Learning object affordances: from sensory-motor coordination to imitation, IEEE Transactions on Robotics 24 (1) (2008) 15–26.
[13] L. Montesano, M. Lopes, Learning grasping affordances from local visual descriptors, in: IEEE 8TH International Conference on Development and Learning, China, 2009.
[14] A. Morales, P.J. Sanz, A.P. del Pobil, Vision-based computation of three-finger grasps on unknown planar objects, in: IEEE/RSJ Intelligent Robots and Systems Conference, 2002.
[15] J. Yan-Bin, On computing optimal planar grasps, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, Pittsburgh, PA, USA, 1995, pp. 427–434.
[16] A. Miller, S. Knoop, H.I. Christensen, P.K. Allen, Automatic grasp planning using shape primitives, in: IEEE International Conference on Robotics and Automation, vol. 2, 2003, pp. 1824–1829.
[17] G. Taylor, L. Kleeman, Grasping unknown objects with a humanoid robot, in: Proc. 2002 Australasian Conference on Robotics and Automation, 2002, pp. 191–196.
[18] R. Horaud, F. Dornaika, B. Espiau, Visually guided object grasping, IEEE Transactions on Robotics and Automation 14 (1998) 525–532.
[19] J. Triesch, J. Wieghardt, E. Mal, C.Y.D. Malsburg, Towards imitation learning of grasping movements by an autonomous robot, in: Proc. 3rd Gesture Workshop, GW'97, in: Lecture Notes in Artificial Intelligence, Springer, 1999, pp. 73–84.
[20] M. Salganicoff, L.H. Ungar, R. Bajcsy, Active learning for vision-based robot grasping, Machine Learning 23 (2) (1996).
[21] J. Pauli, Learning to recognize and grasp objects, Autonomous Robots 5 (1998) 239–258.
[22] T. Baier-Lowenstein, J. Zhang, Learning to grasp everyday objects using reinforcement-learning with automatic value cut-off, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2007, October 29 2007–Nov. 2 2007, pp. 1551–1556.

[23] J. Zhang, B. Rössler, Self-valuing learning and generalization with application in visually guided grasping of complex objects, Robotics and Autonomous Systems 47 (2004) 117–127.

[24] A. Saxena, L. Wong, A. Ng, Learning grasp strategies with partial shape information, in: Proceedings of the 23rd National Conference on Artificial Intelligence, 2008, pp. 1491–1494.

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

[26] J. Tegin, S. Ekvall, D. Kragic, J. Wikander, B. Iliev, Demonstration-based learning and control for automatic grasping, Intelligent Service Robotics (2008).

[27] A. Morales, E. Chinellato, A. Fagg, A. del Pobil, An active learning approach for assessing robot grasp reliability, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2004, 2004.

[28] O. Kroemer, R. Detry, J. Piater, J. Peters, Combining active learning and reactive control for robot grasping, Robotics and Autonomous Systems (2010) 1105–1116.

[29] J. Hilbe, Logistic Regression Models, Chapman & Hall/CRC Press, 2009.

[30] D.W. Hosmer, S. Lemeshow, Applied Logistic Regression, John Wiley and Sons, 2000.

[31] A. Schein, L.H. Ungar, Active learning for logistic regression: an evaluation, Machine Learning 68 (2007) 235–265.

[32] T. Hastie, T. Tibshirani, Generalized Additive Models, Chapman & Hall, 1990.

[33] J. Zhu, T. Hastie, Bayesian beta regression: applications to household expenditure data and genetic distance between foot-and-mouth disease viruses, Journal of Computational and Graphical Statistics (2005).

[34] G. Schohn, D. Cohn, Less is more: active learning with support vector machines, in: Proceedings of the Seventeenth International Conference on Machine Learning, 2000.

[35] S. Tong, D. Koller, Support vector machine active learning with applications to text classification, Journal of Machine Learning Research 2 (2001) 45–66.

[36] M. Sugiyama, S. Nakajima, Pool-based agnostic experiment design in linear regression, Machine Learning and Knowledge Discovery in Databases (2008) 406–422.

[37] A. Krause, A. Singh, C. Guestrin, Near-optimal sensor placements in gaussian processes: theory, efficient algorithms and empirical studies, Journal of Machine Learning Research 9 (2008) 235–284.

[38] J. Bernardo, A. Smith, Bayesian Theory, Wiley & Sons, Chichester, 2000.

[39] B. Schölkopf, A.J. Smola, Learning with Kernels, MIT, 2002.

[40] J. Nocedal, S.J. Wright, Numerical Optimization, second ed., Springer-Verlag, Berlin, New York, 2006.

[41] J. Mockus, On bayesian methods for seeking the extremum, in: Optimization Techniques IFIP TechnicalConference Novosibirs, Lecture Notes in Computer Science 27 (1975) 400–404.

[42] P. Oudeyer, F. Kaplan, V. Hafner, Intrinsic motivation systems for autonomous mental development, IEEE Transactions on Evolutionary Computation 11 (2) (2007) 265–286.

[43] A. Kapoor, K. Grauman, R. Urtasun, T. Darrell, Active learning with gaussian processes for object categorization, in: IEEE 11th International Conference on Computer Vision, 2007.

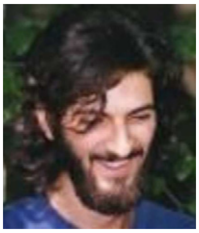
[44] M. Lopes, R. Beira, M. Praça, J. Santos-Victor, An anthropomorphic robot torso for imitation: design and experiments, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'04, Sendai, Japan 2004.

[45] M. Fiala, ARTag, A fiducial marker system using digital techniques, National Research Council Canada, Tech. Report. Code at www.hitl.washington.edu/artoolkit/, 2004.

[46] F.S. Melo, M. Lopes, Learning from demonstration using mdp induced metrics, in: Proceedings of the 2010 European Conference on Machine Learning and Knowledge Discovery in Databases: Part II. ECML PKDD'10, 2010, pp. 385–401.

[47] M. Mason, M. Lopes, Robot self-initiative and personalization by learning through repeated interactions, in: 6th ACM/IEEE International Conference on Human-Robot Interaction, HRI'11, 2011.

Luis Montesano received his Ph.D. degree in computer science in 2006 from the University of Zaragoza, Spain. From 2006 to 2009, he was a Researcher at the Institute of Systems and Robotics (ISR), Lisbon. He is currently an assistant professor at the Computer Science Department of the University of Zaragoza, Spain. He has participated in various international research projects in the areas of mobile robotics and cognitive systems. His research interests include robotics, and machine learning.



Manuel Lopes received the Ph.D. degree in electrical and computer engineering in 2006 from the Instituto Superior Tecnico (IST) Lisbon, Portugal, in the area of robotics. He is a Researcher at the Flowers Group, at INRIA, Bordeau, France. He has participated in various international research projects in the areas of robotics and cognitive systems. His research interests include robotics, machine learning, development and human–robot interaction.

Please cite this article in press as: L. Montesano, M. Lopes, Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions, Robotics and Autonomous Systems (2011), doi:10.1016/j.robot.2011.07.013