González-Díaz, Iván; Benois-Pineau, Jenny; Domenger, Jean-Philippe; Cattaert, Daniel; Rugy, Aymar de (2019) Perceptually-guided deep neural networks for ego-action prediction: Object grasping, *Pattern Recognition*, v. 88, pp.: 223-235.

DOI: https://doi.org/10.1016/j.patcog.2018.11.013

Perceptually-guided Deep Neural Networks for ego-action prediction: Object Grasping

Iván González-Díaz[a], Jenny Benois-Pineau[b], Jean-Philippe Domenger[b], Daniel Cattaert[c], Aymar de Rugy[c]

[a] *Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Leganés, 28911, Madrid. e-mail: igonzalez@tsc.uc3m.es.*
[b] *LaBRI, Laboratoire Bordelais de Recherche en Informatique - UMR 5800 - CNRS Université de Bordeaux, 33405 Talence, France. e-mail: {benois-p,domenger}@labri.fr*
[c] *INCIA - UMR 5287 - CNRS Université de Bordeaux, 33076 Bordeaux, France. e-mail: {daniel.cattaert,aymar.derugy@u-bordeaux.fr*

Abstract

We tackle the problem of predicting a grasping action in ego-centric video for the assistance to upper-limb amputees. Our work is based on paradigms of neuroscience that state that human gaze expresses intention and anticipates actions. In our scenario, human gaze fixations are recorded by a glass-worn eye-tracker and then used to predict the grasping actions. We have studied two aspects of the problem: which object from a given taxonomy will be grasped, and when is the moment to trigger the grasping action. To recognize objects, we using gaze to guide Convolutional Neural Networks (CNN) to focus on an object-to-grasp area. However, the acquired sequence of fixations is noisy due to saccades toward distractors and visual fatigue, and gaze is not always reliably directed toward the object-of-interest. To deal with this challenge, we use video-level annotations indicating the object to be grasped and a weak loss in Deep CNNs. To detect a moment when a person will take an object we take advantage of the predictive power of Long-Short Term Memory networks to analyze gaze and visual dynamics. Results show that our method achieves better performance than other approaches on a real-life dataset.

Keywords: Human perception, grasping action prediction, weakly supervised active object detection

1. Introduction

In this work we tackle the problem of predicting a grasping action in egocentric video. We understand the 'grasping action' as grasping an object of a given category in a complex visual scene which might contain a variety of objects in a cluttered environment. Known paradigms of neuroscience indicate that human gaze expresses intention of a subject and anticipates action [1], which can be helpful for driving recognition of objects in videos with cluttered scenes, and constitute valuable cues about subjects' intention. Hence, using visual data and gaze capturing devices (eye-trackers) can provide deep insights into human perception and behavior as they show what a person is looking at while moving freely in a real-world setting.

The rationale behind this assumption is related to the notion of active perception [2] and intentionality [3], which can be defined as the commitment of a person to perform a particular action [4]. Intention requires skills such as foresight and planning. Perception and action are intimately linked, because a large part of perception is an active process in which the subject anticipates the sensory consequences of its actions. Thereby, perception and generation of behavior can be considered as belonging to the same neural process [5] aiming at testing hypothesis about subjects environment [6]. Among the sensory signals that elaborate perception-anticipation, vision occupies a key position and visual searches might be considered as experiments that generate sensory data. During this search of information, ocular saccades can be viewed and modeled as active experiments that generate the sensory data necessary to perception itself [7]. Although various behavioral traits can be used to catch intention, gaze seems to be preferred because it clearly indicates attention in cooperative tasks where partner gaze is supposed to be the next space to be acted on [8, 9]. Hence, eye gaze can be used efficiently to identify targets during face-to-face conversation [10]. Intent content of gaze was demonstrated to be used in scenarios like collaborative information search in which one sees where the other is looking at [11, 12, 13], gaze-based image retrieval [14] or meal preparation [15, 13]. All

Figure 1: Examples of sequences of geometrically aligned gaze points shown in one frame of the video sequence. Points are labeled in color from the frame in which the user first fixates the object being grasped (red) until the moment before the object is grasped (yellow).

these studies point out the planning information contained in gaze as an active anticipatory process and support our goal of using gaze to anticipate the intention of grasping an object of a given category in egocentric video.

Our target application is assistance to upper-limb amputees wearing neuroprostheses. Although there have been several recent attempts at using computer vision to assist prosthesis control, such as in adjusting a robotic or prosthetic grasp to a recognized object [16, 17], they were typically conducted in very simple visual environments. A critical aspect of our contribution is to enable robust object identification and prediction of grasping actions in more challenging and real visual scenarios, including occlusions and variety of objects of potential perceptual interest present in the scene. Addressing this task appropriately would also be useful in a wide range of applications where identifying the object-of-interest in a scene becomes a key step for subsequent activity recognition [18], video summarization [19] and other visual pattern recognition tasks.

We decompose the prediction of grasping actions into two sequential processes that rely on physiological signals such as gaze or ego-motion. First, we aim to recognize the object to-be-grasped, the 'active object', frame by frame. Second, we use the detection scores together with gaze measurements to identify the moment when a subject aims to grasp this object. For object recognition

we use gaze-driven Convolutional Neural Networks (CNN) as they simulate the capacity of human visual system to focus on particular details in images. Next, we take advantage of the predictive power of Recurrent Neural Networks (RNN), and in particular Long-Short Term Memory (LSTM) cells, to analyze gaze and visual dynamics and detect when the subject will grasp an object.

The use of gaze as a guiding signal for grasping action prediction is very promising. User's foveation helps restricting recognition process to the region-of-interest. For example, in the particular task of active object detection, gaze can be used to efficiently generate weak annotations yet useful to learn object detectors [20]. However, despite these advantages, it remains an open problem as many challenges arise in real environments.

First, a subject aims to grasp one particular object in the scene at a time, and its detection requires to identify which of the many elements in the scene is the 'active' one. This task goes beyond the traditional object detection problem so that object recognition techniques have to be combined with automatic algorithms understanding user perception. Furthermore, in contrast to previous works where the action recognition is done while an object is manipulated [21, 22, 23], in our scenario the subject only intends to grasp objects of interest in an immediate future. Hence he/she does not manipulate them yet. This causes an important loss of visual support as the detection must be performed before actions are actually carried out.

Second, gaze measurements are particularly noisy in our scenario due to several factors. The presence of distractors in a cluttered natural environment yields saccades, peripheral vision is activated to identify objects to grasp, and visual fatigue impacts oculomotor control. Therefore subject's gaze is not always reliably directed toward the object-of-interest. This behavior is much more common when a subject is identifying the object to be grasped than when the object is already being manipulated. This issue is illustrated in Figure 1.

Third, gaze recording provides only partial information about the object location, which is limited to a simple point within the object. It is not sufficient to generate ground truth bounding boxes that are usually required for learning

with CNN architectures, the current dominant paradigm for object detection.

All these observations lead to our weakly-annotated scenario, where the granularity of the annotations moves away from the desired frame-by-frame bounding boxes to the set of clip-level labels and noisy sequences of gaze fixations. In particular, our labels indicate a sequence of frames in which an object is fixated before being grasped in a near future. However, due to the aforementioned factors, it is not ensured that the object is reliably fixated in all frames (see Figure 1). To address this challenging problem, we present several contributions in this paper: a) A gaze-driven detection CNN for objects to be grasped in egocentric video. We train this model using only gaze fixations and weak video clip-level annotations. b) Two alternative methods for noise handling: one that reduces the noise level in gaze data, and another that estimates confidence measures over the gaze points. c) A novel loss to train a LSTM tackling the automatic prediction of grasping actions, which overcomes known limitations of traditional losses in our scenario. d) A new public dataset, *Grasping In The Wild (GITW)* which, in contrast to previous databases, addresses the detection of grasping actions in natural environments before the objects are actually manipulated.

This paper extends the work proposed in [24], which focused on active object recognition. Here, we present the whole system for grasping action prediction, introduce the block to perform the action recognition, describe all modules in detail, and present an extended set of experiments and conclusions. The remainder of the paper is organized as follows: in Section 2 we discuss related works, in Section 3 a detailed description of our approach is provided, in Section 4 we show our experiments and results; finally, conclusions are drawn and perspectives of research are outlined in Section 5.

2. Related work

2.1. Weakly supervised active object recognition

We consider a weak label as an annotation provided at a granularity different than the element to be recognized in a particular task: e.g. learning segmen-

tations using image-level labels that indicate the presence of an object in an image, but do not provide any cue about its location in the scene [25].

Although scene analysis based on weak labels allows reducing the human effort devoted to annotate training datasets [14], it is yet a challenging and open problem due to the distinct granularity of data and labels. In our particular case, whereas object detection networks provide object scores at bounding boxes, our active object labels are given for short video clips, which correspond with those segments of the videos when a subject fixates an object to be grasped. In consequence, aggregation methods are required to bridge the gap between training data and labels. Many works formulate this problem using Multiple Instance Learning (MIL) principles and develop aggregation operators: *avg* [26], *max* [27, 28], *Log-Sum-Exp (LSE)* [26], *global weighted rank-pooling* [29], *negative evidence models* [30], *weighted average* of regions with maximum and minimum scores [31], etc. Other approaches incorporate aggregation methods into novel losses dealing with the different granularities of data and labels: Papandreou et al. [32] use a latent model to generate this loss, Pathak et al. [25] propose a constrained weak loss with inequalities applied over the accumulated probabilities along pixels. In other works, losses are implemented over latent SVMs and the object location becomes a latent variable [33][34].

Detecting the object to be grasped goes beyond traditional object recognition as it also requires to identify which of the objects in a cluttered visual scene is the active one. In this scenario, class-agnostic Region Proposal Networks (RPN) [34, 35, 36], or even specialized CNN proposing category-aware candidate boxes [28], do not perform well; they cannot identify which is the active object among those present in a scene. However, if gaze fixations are available, this physiological signal becomes a valuable cue of the subject intention that might be used to predict the object location. There are methods that, learning from sequences of fixations of subjects observing images, derive bounding boxes that can be used to train object detectors [37, 38]. Closely related to our scenario, some works drive active object detection in egocentric videos using gaze fixations [39] or automatically predicted saliency of pixels [40].

However, none of previous approaches have considered the noisy nature of gaze due to cognitive and physiological factors. Some attempts at reducing noise on the basis of attention models in natural scenes with toy objects have been made in egocentric video, where fragments of initial scene exploration were just removed, and per-frame detection results were filtered in a temporal buffer [1]. We go further, and adapt noise removal approaches to each subject, using his/here recorded data. Confidence measures are also incorporated to a weak loss used in the learning process, with the ultimate goal of learning better visual models for active objects.

2.2. Action prediction

We also aim to predict the exact moment when a subject wants to perform the grasping action. Despite the availability of a very large literature on action recognition in video using 3D CNNs [41] or Recurrent Neural Networks (RNN) [42][43], action prediction has clearly received much less attention and still remains addressed by quite a small number of researchers [44][45]. Perhaps the most related problem that has been thoroughly analyzed is *action anticipation*, in which the observed action has to be identified as soon as possible [46, 47]. Previous works in this task designed specific losses over time to encourage the early prediction of actions [48, 49].

However, our task is even more challenging since the action has not even been started yet. As we do not have strong visual support about the upcoming action, we rely on physiological signals gathered from the subject, such as his visual field, his tracked gaze, and his head/body motion.

3. Proposed model for gaze-driven grasping action prediction

This section describes our model for gaze-driven grasping action prediction in egocentric video. Figure 2 illustrates the steps of the method. It contains three different blocks: a) the *Geometric Alignment* module, described in Section 3.1, estimates and compensates ego-motion to produce normalized gaze points,

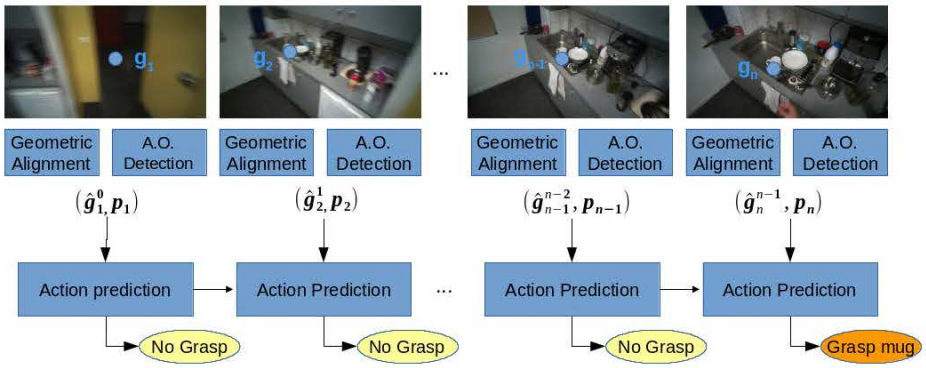

Figure 2: Proposed scheme to predict the action of grasping an object. Our system involves three different building blocks: a) a *Geometric Alignment* module, b) a *Gaze-driven active object detection* module (A.O. Detection); c) an *Action Prediction* module.

b) the *Gaze-driven active object detection* module, introduced in Section 3.2, recognizes what is the active object in the scene; and c) the *Action Prediction* module, detailed in Section 3.4, studies the sequence of gaze points and active object detections to decide when a subject is aiming to grasp an object.

3.1. Geometric Alignment Module

In our egocentric scenario, the subject wears a camera (mounted in his/her glasses). Thus, there is motion between consecutive frames caused by the natural movement of subject's body and head. Even if the subject is looking at the same point in the scene (e.g. the active object), the projected gaze coordinates in the images will vary between two consecutive frames due to ego-motion. Therefore, measured gaze locations in two frames cannot be directly compared unless camera and eye motions have been first decoupled.

To do this, we need to estimate and to compensate the ego-motion between every two consecutive frames. Given a short video clip with N frames and a sequence of gaze points $\mathbf{g}_n = \{(g_{xn}, g_{yn}), n = 1...N\}$, our system operates as follows: for each pair of consecutive frames, it detects and describes local features (SURF features [50] in our case), establishes an initial set of matches, and applies a robust estimation algorithm to compute a projective transformation

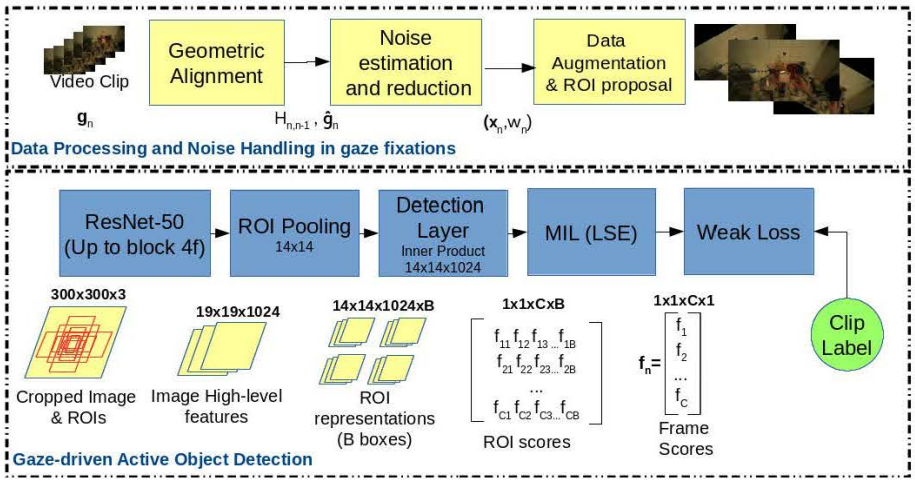

Figure 3: Processing pipeline for Gaze-driven active object detection: a) Data Processing and Noise Handling in gaze fixations, b) Gaze-driven Active Object Detection using weak annotations. Filter sizes are provided when necessary within the layer blocks, in the form of *height* x *width* x *channels in* x *channels out*. Bottom row shows size of the tensors produced by every block, in the form of *height* x *width* x *channels* x *instances*.

$H_{n-1,n}$ (3x3 homography) between frames $n-1$ and n (e.g. DLT algorithm [51] and RANSAC [52]). Since a projective homography only explains transformations of planar surfaces, we restrict the process to a circular area surrounding the gaze point, where depth variations can be neglected without too much error. The radius of this area is initially set to F=100 pixels, and then increased if not enough local features are detected.

Once the sequence of geometric transformations $H_{n,n-1}$ between consecutive frames has become available, we can compute the sequence of aligned fixations $\hat{\mathbf{g}}_n^r$ of every frame n with respect to a reference frame r. This reference frame will vary depending on the part of the method the aligned fixation is used in: during the action prediction task in frame n, the reference r will be the previous frame $r = n-1$; in contrast, for noise handling in weakly supervised object detection, there will be a common reference r, the central frame in each video sequence.

3.2. Gaze-driven active object detection

The processing scheme of the *Gaze-driven active object detection* module is depicted in Figure 3. It is decomposed into two sub-modules: a) a *Data Processing and Noise Handling* block, and b) the *Gaze-driven Active Object Detection using weak annotations*. The first sub-module processes the aligned gaze points provided by the Geometric Alignment module and generates a set of *Regions of Interest* (ROIs) per video frame, candidates to contain the object-of-interest. It also incorporates several processing modules that estimate and reduce the level of noise present in gaze recordings. The second module evaluates the candidate ROIs detecting the object-of-interest along frames of a video clip.

3.2.1. Data Processing and Noise Handling in gaze fixations

The goal of this block is to generate a set of candidate ROIs to search active objects. Hence, two operations are performed sequentially. First, the noise present in the sequence of gaze fixations is processed by one of the two alternatives: 1) reducing fixation noise or 2) estimating a confidence measure associated with each fixation. Then, a set of candidate ROIs is proposed.

Gaze fixations are noisy and quite scattered especially during segments of videos when subjects explore the scene. Since current classifiers are quite sensitive to noise in training data, we perform noise filtering at this phase in order to generate more confident object proposals in each frame.

We consider both the original $\mathbf{g}_n$ and the aligned sequence of gaze points $\hat{\mathbf{g}}_n^r$, where the latter have been computed with respect to a common reference frame r. In this case $r = \lfloor \frac{N}{2} \rfloor$, standing for the central frame of the video clip. Examples of aligned gaze sequences are shown in Figure 1.

The sequence of aligned gaze points is used to predict the most probable location of the active object in a training scene. This was achieved using Kernel Density Estimation (KDE) with 2D Gaussian kernels [53] over the sequence $\hat{\mathbf{g}}_n$ to estimate a 2-dimensional probability distribution modeling the active object location. The global maximum of this distribution becomes our prediction of the object location in the reference frame $\mathbf{o}_r = (o_{rx}, o_{ry})$. Although we tested other

kernels such as the flat kernel, we finally used Gaussian kernels as they provided the best results and are in line with the concept of central and peripheral vision in human retina, whereby human vision resolution decreases as we go away from the gaze fixation. Figure 4 shows two examples of the KDE prediction (in blue).

Considering a training video sequence, our objective is to provide a set of triplets $(\mathbf{x}_n, w_n) = (x_n, y_n, w_n)$ with the weak annotations of the active object location along the frames n. Here x_n, y_n are the coordinates of the points and w_n stands for a confidence measure about the quality of the corresponding point. We have developed two complementary methods to compute these triplets:

1. *Noise Reduction*: with this method, we generate the final sequence of points $\mathbf{o}_n$ in the video clip by applying the inverse accumulated homographies $H_{n,r}^{-1}$ over the predicted object location $\mathbf{o}_r$; $\mathbf{o}_n = H_{n,r}^{-1}\mathbf{o}_r$. Hence, the final set of points used as weak annotations is $\mathbf{x}_n = \mathbf{o}_n$, and $w_n = 1$ for every frame n in the clip. This method strongly reduces the level of noise in the training data as it associates every frame to the same element in the scene (the location predicted by the KDE). Figure 5 depicts two examples of the original sequences of candidate ROIs (left column) together with the ones generated by the method (right column). This example illustrates well the performance of the method.

2. *Estimation of confidence in gaze fixations*: in a second approach, we keep the original captured gaze points in our weak annotations $\mathbf{x}_n = \mathbf{g}_n$ and, alternatively, estimate some weights w_n that represent a confidence in each gaze point. The weights w_n are inversely proportional to the distance between the aligned gaze point $\hat{\mathbf{g}}_n^r$ and the active object location $\mathbf{o}_r$ predicted by the KDE.

$$w_n \propto \exp(-\gamma d(\hat{\mathbf{g}}_n^r, \mathbf{o}_r)) \tag{1}$$

where $d(\cdot)$ stands for the euclidean distance, and γ is a parameter of the model. It has been set to $\gamma = 0.001$ based on some preliminary tests, which is approximately equivalent to normalize the distance between gaze points and object locations by the maximum dimension of the input frame ($W = 960$ in our case). Gaze fixations that, when aligned to the reference frame, are close to the predicted location, are considered as 'good' samples. They receive high weights,

Figure 4: Examples of object prediction using KDE over aligned gaze points. Aligned gaze points are in red to yellow according the time stamp, predicted object location is in blue.

whereas points situated far from the object location receive lower weights. In this second approach, rather than explicitly removing noise (we are keeping the original gaze points), we are estimating a confidence measure that will be used to train our model for active object detection (see Section 3.3). The rationale behind this alternative is that keeping the original gaze points may provide more visual diversity than simply generating all the annotation points as projections of the same location.

During testing phase the original gaze fixations $\mathbf{x}_n = \mathbf{g}_n$ without confidence and without filtering are considered, as the algorithm performs online, on a frame-by-frame basis.

We also perform *data augmentation* rotating input images using angles in the range $[-45, 45]^\circ$. These rotations imitate subjects' views when approaching the active object from different angles/locations in the scene. Preliminary experiments have shown that data augmentation provided an improvement of about 2-4% in performance. In addition, for each image and annotation point, we adapt to the varying scale and shape of objects, proposing regions-of-interest (ROIs) at 3 scales $s = (78, 125, 200)$ px and 3 aspect ratios $r = (0.8, 1, 1.25)$. We therefore use $B = 9$ bounding boxes of size $w \times h$, with $w = s/\sqrt{r}$ and $h = s\sqrt{r}$, centered on random shifts around the annotation point $\mathbf{x}_n$ of the frame. This particular number of bounding boxes yielded good performance in some preliminary experiments and is similar to the number of anchors used in reference approaches for object detection [54][55]. Figure 5 shows several examples of our ROI candidate set.

Figure 5: An illustration of Noise Reduction technique. (Left) Frames with candidate ROIs computed over the original sequence of gaze points. (Right) Noise-free candidate ROIs by back-projecting the predicted object location to the frame sequence.

3.3. Gaze-driven Active Object Detection using weak annotations

This block aims to perform a robust detection of active objects using gaze as a weak and noisy guiding signal. The system is depicted in the bottom row of Fig. 3 and involves several building blocks. It builds over the well-known ResNet- 50 CNN [56]. Residual networks do not learn the desired underlying mapping at each layer, but rather fit a nonlinear residual mapping by summing it to a direct linear connection between inputs and outputs. Hence, residual layers avoid the degradation problem when additional layers are stacked to the network, as they easily learn zero mappings when no additional transformations are required. This is why the residual networks have become very popular.

Among all the ResNet variations (ResNet-18, 50, 101 and 152), we found that a ResNet-50 yielded very competitive results at acceptable computational times. In particular, we incorporated several new layers at the top of ResNet-50 to perform the object detection in our weakly-annotated scenario. For the sake of

completeness, we will introduce all these new layers, but our main contribution here is the weak loss described in Section 3.3.3.

The network is initialized with the original weights up to the layer *res4f*, being the remaining layers discarded. Our own blocks are stacked instead to enable object detection with weak annotations. Furthermore, we use Batch Gradient Descent algorithm for training, where each batch contains frames of the same video clip, which allows to learn from labels at a video clip level.

3.3.1. ROI pooling and detection layers

The output of the ResNet-50 subnet is a high-level visual representation of the input image. In practice, for an input crop of 300×300 (centered on the gaze point) we generate a tensor of size $19 \times 19 \times 2048$, with a reduced spatial dimension (19×19) and an extended set of 2048 high-level feature channels. A nice consequence of our approach is that all this processing is made just once as it is common for every evaluated ROI in the frame.

Then, we need to generate individual representations for each considered candidate ROI, for which we use the efficient ROI Pooling method described in [57], and produce a set of B $14 \times 14 \times 2048$ tensors associated to each of the candidate ROIs. Once we have an independent representation of each ROI, we can compute a detection score for each object category. With this purpose, the so-called *detection layer* is a fully-connected layer that transforms the $14 \times 14 \times 2048$ input tensor into a C-length vector $f_{c,b}$ with the scores of the different classes c in the ROI b. Finally, concatenating the vectors of all ROIs, we can generate the matrix of ROI scores $\{f_{c,b}\}$ shown in Figure 3, where c stands for the object class and b for the bounding box.

3.3.2. MIL aggregation

Given the matrix with the ROI scores $\{f_{c,b}\}$, we generate a vector of frame-level predictions using Multiple Instance Learning (MIL), assuming that there exists at least one candidate ROI corresponding to the active object. In particular, we have used Log-Sum-Exp (LSE) aggregation proposed in [26]. Given

a set of class scores $f_{c,b}$ for the B ROIs in a frame, the *aggregated score at the frame level* f_c is computed as:

$$f_c = \log\left(\frac{1}{B}\sum_{b=1}^{B}\exp\left(f_{c,b}\right)\right) \tag{2}$$

The aggregation will produce the vector of frame-level scores for the object categories. This vector can be finally transformed into a set of *object probabilities* using a simple softmax operator: $p_c = softmax(f_c)$.

3.3.3. Weak Loss

Our training process requires an additional step as the weak labels indicating the presence of an active object are given for each short video clip. Since we cannot ensure that the object to be grasped is being fixated at every frame, we do not have annotations at frame level, and therefore need an additional layer aggregating the frame scores at the video level, and a loss function stacked at the top of our network, that compares the video scores with the video labels.

To cope with all these requirements, we have adopted a constrained loss for learning under weakly annotated scenarios [25], which was initially developed for pixel-wise weak semantic segmentation. In our work, instead, it has been adapted for active object detection in weakly annotated video, defining constraints over the accumulated scores along frames, and incorporating the confidence weights w_n. Considering C object categories in our problem, marginal independence between frames, the probability distribution of a video clip with N frames can be factorized as:

$$P(\mathbf{c}|\theta) = \prod_{n=1}^{N} p(c_n|\theta)^{w_n} \tag{3}$$

where $\mathbf{c}$ is a random variable of the active object classes, θ stands for the parameters (weights) of the CNN, $p(c_n|\theta) = p^n_{c_n}$ stands here for the class probabilities c_n in the frame n, and w_n is the weight associated with each training frame in the video (as defined in Section 3.2.1). This leads to an optimization problem with inequality constraints:

$$\text{find} \quad \theta; \quad \text{subject to} \quad A\overrightarrow{P} \geq \overrightarrow{b} \tag{4}$$

where $\overrightarrow{P}$ is the vectorized version form of the network output $P(\mathbf{c}|\theta)$, and $A \in \mathbb{R}^{K \times C \cdot N}$ and $\overrightarrow{b} \in \mathbb{R}^K$ define K linear constraints over the output distribution P. Since this problem is not convex with respect to the network parameters θ, the authors in [25] defined a variational latent probability distribution $Q(\mathbf{c})$ over the object categories (independent from the CNN parameters θ), applied the constraints to this new distribution rather than to the original network output $P(\mathbf{c}|\theta)$, and enforced $Q(\mathbf{c})$ to be similar to $P(\mathbf{c}|\theta)$ by minimizing their Kullback-Leibler divergence. The interested reader is referred to the original paper [25] for an in-depth derivation of the equations involved in the learning process (let us note that the set of weights w_n was not included in the original approach).

By setting proper values of the parameters A and $\overrightarrow{b}$, we can impose constraints over the object class probabilities along each video clip. In particular, given a category c, we impose the following constraint:

$$K_{c,min} \leq \frac{1}{\sum_n w_n} \sum_{n=1}^{N} w_n p_c^n \leq K_{c,max} \tag{5}$$

Here p_c^n is the probability score of the class c in the frame n, $K_{c,min}, K_{c,max} \in [0, 1]$ are model parameters that stand for the minimum and maximum percentage of accumulated probability that corresponds with the category c in the video segment, respectively. Intuitively, we are imposing that the percentage of the frames that show the object c in the fixation area should range between $K_{c,min}$ and $K_{c,max}$. Setting appropriate values for these parameters allows to deal with the presence of noise, and the existence of frames showing background or even other non-active objects. In particular, we have validated these parameters and found that the following values yielded reasonable results: a) if c represents the *active object* in a video clip, then $K_{ao,min} = 0.85$, and $K_{ao,max}$ is deactivated; b) for those classes c that are considered as *non-active* in a video segment, $K_{nao,max} = 0.01$, whereas $K_{nao,min}$ is deactivated; and c) for the *background class* $c = 0$, $K_{0,min} = 0$ and $K_{0,max} = 1 - K_{ao,min} = 0.15$.

These values deserve a discussion: $K_{ao,min}$ is dependent on the dataset, as it models the percentage of time a subject fixates the object, and is closely related

to the probability of finding gaze deviations from the active object. Hence, the presence of very cluttered scenes with many potential distractors would lead to lower values of the parameter whereas simple scenarios in which the active object is not surrounded by any other element would probably accept larger values closer to 1. For non active objects, the maximum accumulated probability $K_{nao,max}$ has to be low as the object should not be regularly fixated. We found that very low but non-zero values ($K_{nao,max} = 0.01$) provided the best results. With respect to the background class, $K_{0,max}$ has been automatically set from the value $K_{ao,min}$ so that this parameter does not need an individual validation.

3.4. Grasping Action Prediction

The last block of our system is the *Grasping Action Prediction* module shown in Figure 2. It uses aligned gaze points provided by the Geometric Alignment module (Section 3.1) and the active object probabilities computed by the Gaze-driven Active Object Detection module (Section 3.3), to detect when a subject is aiming to grasp an object. We define this problem as a multi-class classification problem, with C+1 action classes, $c = 0$ standing for the class *'No grasp'*, and $c = 1...C$ for the actions *'Grasp the object of class c'*. This multi-class formulation of the problem allows us to temporally filter the scores of the active object detector and better account for potential subject's gaze deviations.

As in a real test scenario we do not have future information to perform noise filtering over the gaze fixations (as we do during training of object detectors), we propose to use Recurrent Neural Networks and, in particular, Long-Short Term Memory cells [58, 59] to perform a temporal filtering of the data. For the sake of conciseness we will not include a detailed description of the equations that govern the LSTM; the interested reader is referred to the work of Graves et. al [60] for an in-depth explanation of LSTMs in deep neural networks. We feed the LSTM with a shallow concatenation of four features:

a) *Magnitude of the gaze motion vector*: It is computed over gaze points of consecutive frames, where the new gaze point is first aligned to the previous frame to remove the ego-motion: $\mathbf{u}_n = \hat{\mathbf{g}}_n^{n-1} - \mathbf{g}_{n-1}$. Gaze motion is a

physiological measure that becomes a valuable cue about the user's intention. It helps distinguishing between different kinds of subject's interaction with the environment: scanning the scene, identifying the object to be grasped, starting the grasping action, etc.

b) *Magnitude of the ego-motion*: modeling the subject's head and body movement might also be useful to understand subject's interaction with the environment. This magnitude is computed by subtracting the gaze-motion vector from the total displacement of the gaze point: $\mathbf{v}_n = \mathbf{g}_n - \mathbf{g}_{n-1} - \mathbf{u}_n = \mathbf{g}_n - \hat{\mathbf{g}}_n^{n-1}$.

c) *Distance of the gaze point to the center of the image*: This feature was added on the basis of our observations on the so-called 'central bias' hypothesis. When looking at a still image or video, a subject first foveates near the center of the frame. Furthermore, in ego-centric setting, the subject adapts his/her body and head pose in such a way that the active object is in the center of the frame.

d) *Vector of active object scores* for the frame $\mathbf{p}_n$, as including the sequence of active object detections is fundamental to trigger the grasping action.

The particular architecture of our network is as follows: for each frame in a video, the aforementioned input vector feeds a bottom LSTM layer with 256 units. The hidden state cells of this last layer are finally passed to a fully connected layer that produces the output $\mathbf{s}_n \in \mathbb{R}^{(C+1)\times 1}$ of the system: the vector of scores of the C+1 considered actions. This vector can be in turn converted into a vector of probabilities using a softmax layer $\mathbf{y}_n = softmax(\mathbf{s}_n)$. In addition, during learning we incorporate a *dropout layer* with a factor of 0.5 to reduce over-fitting. Similar architectures have been successfully applied to action recognition [42].

3.4.1. A novel loss for action prediction

In many multi-class problems, the *cross-entropy loss* is used in training. It is the cross-entropy between the action label distribution A_n and the system output $\mathbf{y}_n$. For a given frame n, and considering C+1 action classes, the label

A_n is a $(C+1)\times 1$ index vector with a 1 in the position of the true class label a_n and zero elsewhere. Hence, for a video sequence of N frames, the cross-entropy loss is as follows:

$$\mathcal{L}(\mathbf{y}, \mathbf{A}) = -\sum_{n=1}^{N} \log y_n(a_n) \tag{6}$$

where $y_n(a_n)$ is the probability of the true class a_n.

However, in our particular problem of action prediction, this general loss shows two limitations:

1. If during training, we find sequences in which users hardly fixate the active object (e.g., if they use peripheral vision all around the segment of interest, before the object is grasped), the sequence of active object scores is wrong, and the LSTM trained with cross-entropy loss tends to revert the situation by learning a wrong mapping between object and action scores. We observed that this anomaly leads to over-fitting and, as we will show in the experimental section, strongly limits performance in test. Thus, we would like to enforce some degree of alignment between the detected objects and the predicted actions.

2. To reduce the impact of the system on the subject's natural behavior, we would like to successfully predict a grasping action early on, even if users only fixate the active object during short temporal segments. Hence, we would like a loss function that encourages early predictions of actions.

To satisfy these two requirements, and following similar procedures in the field of early action detection [49], we have developed a new loss function:

$$\mathcal{L}(\mathbf{f}, \mathbf{A}) = -\sum_{n=1}^{N} \log y_n(a_n)^{p_n(a_n)\tau_n} \tag{7}$$

where $p_n(a_n)$ is the probability given by the gaze-driven active object detection that the object corresponding to the true class a_n is the active one, and τ_n is a weight of *temporal importance*. Considering that n_{fix} is the frame in which a user starts to fixate the active object, we define τ_n as:

$$\tau_n = \begin{cases} 1, & n \leq n_{fix} - N_p \\ \frac{n_{fix}-n}{N_p}, & n_{fix} - N_p < n \leq n_{fix} \\ 1, & n > n_{fix} \end{cases} \tag{8}$$

where N_p stands for the number of frames, previous to the annotated start of active object fixation, during which our learning procedure penalizes less deciding the grasping action. This value has been heuristically set to 10 frames. Intuitively, our loss function behaves as follows: during the segment before the subject fixates the object of interest $n < n_{fix} - N_p$, $\tau_n = 1$ and the network completely learns the label 'no grasping action'. However, N_p frames before the subject fixates the object and, due to the reasons mentioned above, the learning weight starts at $\tau_n = 1$ and decreases linearly until the exact moment when the active object is fixated $n = n_{fix}$. This means that the classification errors in training are less important during this short segment. Finally, during active object fixation, the learning weight is again $\tau_n = 1$ and the network completely learns again the action 'grasping the object c'.

Hence, with this new loss, we are decreasing the influence of two kinds of frames during the learning phase: a) frames in which a user is not looking at the active object ($p_n(a_n)$ will be small), causing a wrong score in the active object detector, and b) frames that occur just before the subject starts to fixate the object of interest.

4. Experiments and results

In this section we present our experiments and results for grasping action prediction. We will first assess the performance of the module of gaze-driven active object detection, focusing on the impact of our proposed noise handling techniques. Then, the experiments carried out using the system for gaze-driven grasping action prediction will be described.

4.1. Dataset and experimental set-up

We run our experiments using a new recorded egocentric dataset: *Grasping In The Wild (GITW)*[1]; which was designed and recorded because all previous

[1] www.labri.fr/projet/AIV/dossierSiteRoBioVis/GraspingInTheWildV2.htm

databases focused on the detection of active objects during their manipulation [39, 21, 22] and did not fit with our goal of to-be-grasped object detection.

GITW has been recorded with Tobii Glasses 2 worn by subjects performing activities of everyday life in an ecological environment (7 kitchens). It contains 404 egocentric videos of lengths varying between 3.5 and 26 seconds, with a total length of 62 minutes. Videos have been recorded with a resolution of 1920x1080 at a frame rate of 25 fps, whereas gaze points were acquired at 50 Hz. As we have two gaze recordings per frame duration, the gaze fixation point associated with each frame has been computed using spline interpolation based on previous and current gaze recordings.

The dataset contains 16 categories of objects to be grasped, which correspond to objects often found in a kitchen: bowl, can of coca-cola, frying pan, glass, jam container, pan lid, milk container, mug, oil bottle, plate, rice container, sauce pan, sponge/scourer, sugar container, vinegar bottle, and washing up liquid. A maximum of 4 different subjects perform activities in each kitchen. The recording protocol was as follows: Each subject first listened to the instruction with the name of the object-to-grasp. Hence the subject explored a visual scene to find the location of target object and finally grasped it. We have annotated the dataset labeling the temporal segment starting when the user fixates the active object and ending at the instant just before the object is grasped. This segment corresponds with the moment during which we aim to detect the object to-be-grasped. GITW dataset is very challenging because active objects should be recognized before they are actually manipulated (the subject is just looking at them) and scenes usually contain many objects that are not active.

We have divided our dataset into 5 folds of 80-81 video clips and around 18700 frames each, out of which 2290 correspond to the temporal segment of interest. In order to obtain statistically significant results, we have followed a 5-fold cross-validation approach, repeating the experiments 5 times and leaving each time one fold for test. In addition, we decreased the initial resolution of the videos by a factor of 2, working with frames of size 960x540.

Table 1: Results of our proposals and several compared methods in GITW dataset, given in mean Average Precision (mAP) and Accuracy (mAcc) % with standard deviations. In addition, p-values comparing references algorithms with our GDOD-COMB method are given.

Type	Algorithm	mAP ± std (p-value)	mAcc ± std (p-value)
Strongly supervised	Strong Ours	75.0 ± 1.8 (0.01)	68.4 ± 3.9 (0.02)
	DEEPMASK	67.0 ± 5.2 (< 0.01)	60.5 ± 6.4 (< 0.01)
	YOLO	63.8 ± 6.5 (< 0.01)	62.0 ± 5.0 (< 0.01)
Weakly supervised	MIL-AVG	72.2 ± 3.1 (< 0.01)	66.7 ± 6.1 (0.03)
	MIL-LSE	78.0 ± 3.3 (0.15)	71.0 ± 2.5 (0.06)
	CCNN	78.2 ± 2.8 (0.14)	69.4 ± 3.8 (0.04)
Our method	GDOD-CE	80.5 ± 4.3	73.5 ± 4.4
	GDOD-NR	81.6 ± 2.5	74.7 ± 2.9
	GDOD-COMB	$\mathbf{81.9 \pm 3.7}$	$\mathbf{75.3 \pm 3.3}$

4.2. Results for active object detection

In this section, we assess the performance of our gaze-driven and weakly-supervised object detection method. Our goal is not yet to detect when an object has to be grasped but to simply identify which is the active object in each video sequence. In order to do so, we compare proposed method with several reference methods that can be decomposed into two blocks: First, we have included methods for strongly supervised object detection:

1. Baseline Strong (Strong): a strongly supervised classifier that considers each candidate ROI in every frame a positive sample of the active object.
2. Deepmask [36]: Deepmask is a Region Proposal Network (RPN) that generates a set of candidate boxes with high objectness scores. Those candidate boxes proposed by Deepmask that contain the gaze point of the frame are considered positive samples. If several candidate boxes contain the gaze point, MIL-LSE is used to generate frame-level scores.
3. YOLO [55]: It is a state-of-the-art method for object detection which is particularly suited for our real-time scenario due to its low computational complexity. YOLO integrates proposal of candidate boxes and evaluation of object detections. During training, the B=9 ROIs described in

Subsection 3.2.1 are taken as strong samples and used to set the network anchors. Then, during test, the network is in charge of both proposing the candidate ROIS and evaluating the presence of the objects in them.

In addition, we included methods used in weakly annotated object detection:

1. Multiple Instance Learning methods: including Average aggregation (MIL-AVG), and Log-Sum-Exp aggregation (MIL-LSE) [26]: used both to generate frame and video-level outputs from ROI scores. We did not include Max aggregation [27, 28] as it yielded poor performance in our scenario.
2. Constrained CNN (CCNN) [25]: a constrained weak loss over the video accumulated object probabilities.

Finally, we included three versions of our *Gaze-Driven Object Detection (GDOD)* approach in the comparison:

1. GDOD-NR: that employs our Noise Reduction (NR) mechanism.
2. GDOD-CE: which uses our Confidence Estimation (CE) method.
3. GDOD-COMB: a fusion of the previous ones, combining the ROIs and weights proposed by GDOD-NR and GDOD-CE.

To establish a fair comparison, we use gaze points to guide the detection process of every method. In addition, all methods but YOLO employ the same detection network, except of the noise handling block, the aggregation methods and the learning losses, which differ between approaches.

We have used *mean Average Precision* (mAP) and *mean Accuracy* (mAcc) as our performance metrics, computed over all the object categories (after background class removal). Video clip labels were used as the ground truth, which implies that, even in test, we need to aggregate our scores to produce a unified value for the whole video clip. This contrasts with the action detection evaluation of the next section, where we will use frame-level annotations.

Results of our experiments on GITW dataset are shown in Table 1. The conclusions are as follows. First, region proposal methods such as Deepmask [36] or traditional strongly-supervised object detection methods as YOLO [55],

fail in our problem. When the gaze does not point to a particular object but to intermediate areas, they tend to propose boxes over non-active objects, thus confusing the learning algorithms. Then, among aggregation methods, LSE achieves better results than AVG. This is due to the apparition of distracting elements in the scene, which guide gaze to spatial areas without active objects. Simple average including these frames decreases performance. In contrast, LSE aggregates scores considering those frames with highest values (those in which the active object is fixated), and therefore yields better performance. The weak loss proposed in CCNN [25] also considers the presence of distractors, providing similar results to LSE and outperforming the strongly-supervised approaches.

However, none of these weak learning techniques explicitly handle noise in the way we propose in this work, analyzing the whole video sequence and estimating the most probable area to contain the active object. Hence, we can see in the table that our two proposed techniques for noise handling improve the performance of any other compared method in both mAP and mAcc. Furthermore, although individual results show that reducing noise in gaze fixations might be preferable than using confidence measures over the original gaze points, our combined version (GDOD-COMB) takes the best of the two approaches and achieves the top performance on our dataset. Regarding the consistency of our results, for all the compared algorithms the standard deviation values are low compared to the average performances. In addition, in Table 1 we also provide p-values of every reference method computed with respect to our GDOD-COMB approach. With this value we measure the probability that results achieved by two methods come from two distributions with the same means. Despite the statistical limitations of a 5-fold cross-validation, p-values demonstrate that our improvements are significant with respect to every compared method in mAcc, and almost every method in mAP.

In addition to the average results discussed above, in Figure 6, we include the accuracy confusion matrix of the GDOD-COMB version of our approach. From the figure, we cannot see any particular correlation in the errors (e.g. an object category that is often confused with another one). In fact, after observing the

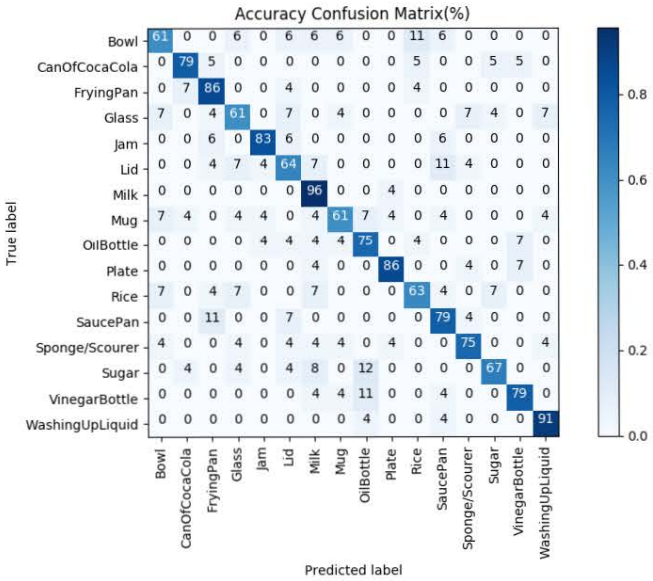

Figure 6: Accuracy Confusion Matrix (%) of the GDOD-COMB version of our approach.

videos, most errors are made between objects that often appear closely located in some of the recorded kitchens regardless of their categories. Our approach thus successfully discriminates between objects, and the main source of errors is the noise associated with the gaze recordings (e.g. our system might predict that a distracting object gathering some gaze points is the one to be grasped).

We have also studied those video sequences that can be considered as hard cases. Some examples are displayed in Fig. 7. We show three cases in which our system successfully detects the object to grasp (left) and other three where it makes errors. Successful examples illustrate that, even if the object is not fixated all the time and gaze may be partly directed to distractors, our system is able to automatically detect the right object as long as it is fixated for a longer time than the remaining elements. However, when more fixations are located at another object (see right examples), our system detects a wrong object to grasp. To get some insight on the limitations of our approach, it is worth identifying when this situation happens: sometimes, there are several distracting objects located very close to the active one (plate is very close to the

Figure 7: (Left) Examples with successful predictions: from top to bottom: milk container, mug and jam container. (Right) Examples in which our system predicts a wrong class. From top to bottom we list predicted/real class: plate/mug, frying pan/sauce pan and frying pan/can of coca-cola.

mug in the top-right image), other times the object is dramatically occluded by another (e.g. the frying pan located over the sauce pan in middle-right image) and, finally, we have observed cases, mainly on clean areas of the scene, in which the subject uses peripheral vision to grasp an object (e.g. the can of coca-cola in the bottom-right image is never fixated).

Finally, we have also evaluated if all frames during the segment of interest are equally informative about the object to grasp. Our initial intuition was that the frames close to the moment of grasping are perhaps more stable and would provide better performance. Results are presented in Figure 8 and show the performance of our best solution GDOD-COMB as a function of the time

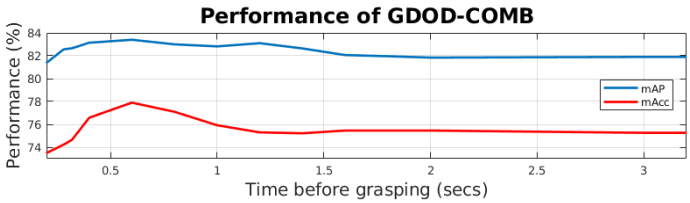

Figure 8: Performance of proposed gaze-driven active object detection as a function of the time before grasping considered during test

before grasping considered during test. The length of the fixation segments varies in our dataset from 280 ms to 3 seconds. Although using the whole video segments is very useful during training, as we include frames showing active objects under different viewpoints and scales, optimal active object detection results ($mAP = 83.4$ and $mAcc = 77.9$) are obtained when considering the last 600ms before grasping. Hence, we can conclude that frames are more valuable as they are closer to the final moment at which a subject is grasping the object.

4.3. Results for grasping action prediction

In this section, we evaluate the performance of the global system for grasping action prediction. In this case, the evaluation is based on the annotations of temporal segments in which the user fixates the object before grasping it. Our goal is to detect the grasping action within those segments.

We evaluate our system using F-score $F(\Delta t)$ computed at different times $\Delta t = t - t_{fix}$, where $t_{fix} = n_{fix}/Fr$ is the time stamp indicating when a user starts to fixate the object, and $Fr = 25$ Hz. F-score is a common metric to evaluate detection that joins precision and recall into one measure as follows:

$$F(\Delta t) = 2\frac{Prec(\Delta t) \cdot Rec(\Delta t)}{Prec(\Delta t) + Rec(\Delta t)} \tag{9}$$

where $Prec(\Delta t)$ and $Rec(\Delta t)$ are the precision and recall measures computed at the times Δt, respectively.

In addition, we consider two problems being addressed: first, the multi-class detection evaluation with C+1 classes ('no grasp' and C classes indicating 'grasp the c-th object'); second, a binary detection problem in which we are only concerned about detecting the grasping action at the right moment.

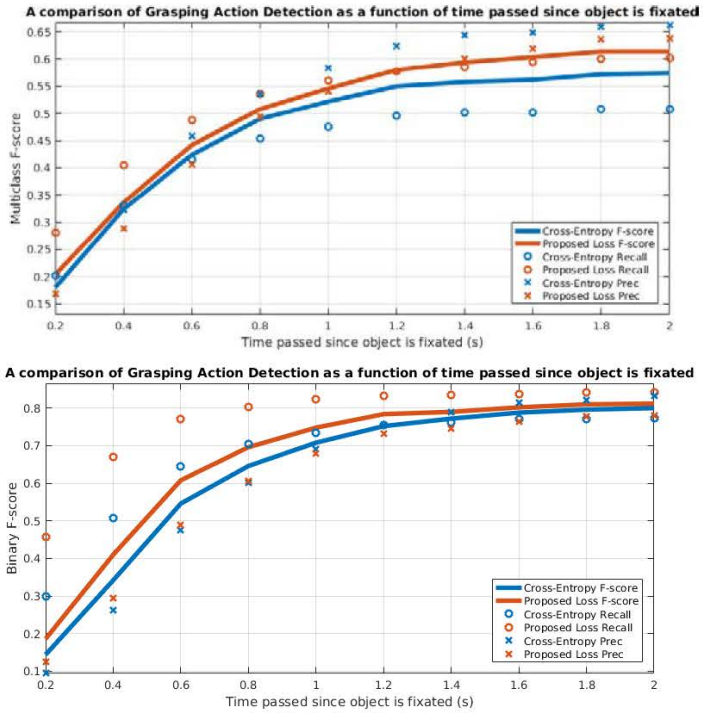

Figure 9: Performance of our grasping action detection as a function of Δt, the time passed since the object is being fixated. (Top) Performance of a multi-class problem with C+1 classes. (Bottom) Performance of an object agnostic binary problem grasp/no grasp.

Results from both experiments are shown in Figure 9 in terms of $F(\Delta t)$. For the sake of completeness, we also provide Recall and Precision measures that gave place to each particular F-score. We have compared the performance of a baseline detection network trained with a Cross-Entropy Loss with our proposed loss. We can see that our novel loss outperforms the baseline due to two reasons: first, it avoids over-fitting by decreasing the learning weights of cases in which object scores and object labels disagree, and second, it encourages detecting the action as early as possible once the object has been fixated. Indeed, the results of the binary classification problem show that the relative improvement of our proposal is larger at low values of Δt, which validates our approach. In the multi-class problem, the effect of the temporal weights is less evident as other factors such as detecting the right object affect the F-score.

Table 2: Analysis of the execution times of different steps in our algorithm during test

Block	Time (ms)
Geometric Alignment	8.8 ms
Object Detection	25 ms
Action Detection	0.8 ms
Total	34.6 ms

The difference in performance between the two evaluations demonstrate that most errors in our system come from sequences in which subjects use peripheral vision to perform the action and do not directly fixate the object to be grasped.

4.4. Assessment of the method in the target application

a)*Evaluation of perception-driving hypothesis*: Our method is based on the assumption that the human looks at the object to grasp; thus we incorporate the knowledge from neuroscience on visual anticipation of an action. Hence it is interesting to evaluate the validity of this hypothesis when human observers take decisions. We have asked 6 subjects to predict the active object looking at the sequence of aligned gaze points shown in Figure 1. We obtained the following results in average and standard deviations: $\mathbf{mAP} = \mathbf{81.7 \pm 1.7}$ and $\mathbf{mAcc} = \mathbf{78.2 \pm 2.6}$, a performance very similar to our automatic method (see optimal performance in Figure 8). This result shows that identifying the active object is not a simple task for a human when the object is not yet manipulated, and validates the usefulness of our automatic approach. Hence, looking and fixating the object of interest is a sine qua non condition for the perceptually guided recognition. This hypothesis is quite reasonable for our target application of assistance to upper-limb amputees wearing neuro-prostheses.

b) *Computational times*: We have measured execution times for every step necessary during test and obtained the following figures presented in Table 2, running our detection system using a NVIDIA TITAN X GPU takes 34.6 ms per frame. Let us note that we are using CUDA implementations of OpenCV 3.4 and Caffe learning frameworks. Hence, our method is very efficient and meets our real-time requirements of 25 fps.

5. Conclusions and Future Work

We have presented a system for perceptually-guided prediction of grasping actions with the ultimate goal of automatic control of prosthetic arms. We exploit physiological signals such as gaze information produced by visual search and used the hypothesis from neuroscience that humans anticipate actions by gaze. Our scenario is particularly challenging as predicting actions before they are actually carried out introduces many problems, including loss of visual support as objects are not manipulated, as well as several artifacts/noise in gaze sequences related to the physiology of human perception.

We have developed a method that first identifies the objects of interest on a frame-by-frame basis, and then uses this information together with captured physiological signals (gaze movements, position and head/body ego-motion) to predict the upcoming grasping action.

Our scenario for active object detection included weak and noisy labels. Hence, we have designed methods that estimate and reduce noise present in gaze sequences, and incorporated them to a weakly-labeled active object detection system. We showed that our methods achieve better performance than other reference approaches on a dataset specifically designed for our target application.

Furthermore, we have proposed a novel loss over RNNs to perform grasping action prediction, which addresses two drawbacks of traditional losses in our scenario: 1) wrong predictions of object detection block may lead to over-fitting during training, and, 2) they do not encourage early predictions of the grasping action, a key factor to optimize system usability. Our experiments have demonstrated that our loss successfully addresses both issues and enhances the system performance by a $2-5\%$ with respect to a baseline method trained using the cross-entropy loss function.

Our future lines of research will focus on the study of multitask models that concurrently address action prediction and gaze forecasting using shared latent variables that encode the state of the visual dynamics.

6. Acknowledgments

This work was partially supported by French National Center of Scientific research with grant Suvipp PEPS CNRS-Idex 215-2016, by French National Center of Scientific research with Interdisciplinary project CNRS RoBioVis 2017-2019, the Scientific Council of Labri, University of Bordeaux, and the Spanish Ministry of Economy and Competitiveness under the National Grants TEC2014-53390-P and TEC2014-61729-EXP.

References

[1] P. P. de San Roman, J. Benois-Pineau, J.-P. Domenger, F. Paclet, D. Cataert, A. de Rugy, Saliency driven object recognition in egocentric videos with deep cnn: toward application in assistance to neuroprostheses, Computer Vision and Image Understanding 164 (2017) 82 – 91.

[2] J. J. Gibson, The Ecological Approach to Visual Perception, Houghton Mifflin, 1979. ISBN: 978-1-848-72578-2.

[3] M. Bratman, Intention, plans, and practical reason, Harvard University Press, Cambridge, MA, 1987. ISBN: 978-0-674-45818-5.

[4] B. F. Malle, J. Knobe, The folk concept of intentionality, Journal of Experimental Social Psychology 33 (1997) 101 – 121.

[5] R. Möller, Perception Through Anticipation. A Behaviour-Based Approach to Visual Perception, Springer US, Boston, MA, 1999, pp. 169–176.

[6] R. L. Gregory, Perceptions as hypotheses, Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences 290 (1980) 181–197.

[7] K. Friston, R. Adams, L. Perrinet, M. Breakspear, Perceptions as hypotheses: Saccades as experiments, Frontiers in Psychology 3 (2012) 151.

[8] S. BaronCohen, S. Wheelwright, J. Hill, Y. Raste, I. Plumb, The reading the mind in the eyes test revised version: A study with normal adults, and

adults with asperger syndrome or highfunctioning autism, Journal of Child Psychology and Psychiatry 42 (2001) 241–251.

[9] A. N. Meltzoff, R. Brooks, 'like me' as a building block for understanding other minds: Bodily acts, attention, and intention, in: Intentions and Intentionality: Foundations of Social Cognition, MIT Press, 2001, pp. 171–191.

[10] J. E. Hanna, S. E. Brennan, Speakers eye gaze disambiguates referring expressions early during face-to-face conversation, Journal of Memory and Language 57 (2007) 596 – 615.

[11] A. Kendon, Some functions of gaze-direction in social interaction, Acta Psychologica 26 (1967) 22 – 63.

[12] S. E. Brennan, X. Chen, C. A. Dickinson, M. B. Neider, G. J. Zelinsky, Coordinating cognition: The costs and benefits of shared gaze during collaborative search, Cognition 106 (2008) 1465 – 1477.

[13] C.-M. Huang, S. Andrist, A. Saupp, B. Mutlu, Using gaze patterns to predict task intent in collaboration, Frontiers in Psychology 6 (2015) 1049.

[14] S. Lopez, A. Revel, D. Lingrand, F. Precioso, Handling noisy labels in gaze-based CBIR system, in: Advanced Concepts for Intelligent Vision Systems - 18th International Conference, ACIVS, 2017, pp. 396–405.

[15] W. Yi, D. Ballard, Recognizing behavior in hand-eye coordination patterns, International Journal of Humanoid Robotics 06 (2009) 337–359.

[16] J. Weisz, P. Allen, A. G Barszap, S. S Joshi, Assistive grasping with an augmented reality user interface, The International Journal of Robotics Research 36 (2017) 543–562.

[17] M. Markovic, H. Karnal, B. Graimann, D. Farina, S. Dosen, Glimpse: Google glass interface for sensory feedback in myoelectric hand prostheses, Journal of Neural Engineering 14 (2017) 036007.

[18] I. González-Díaz, V. Buso, J. Benois-Pineau, G. Bourmaud, R. Mégret, Modeling instrumental activities of daily living in egocentric vision as sequences of active objects and context for alzheimer disease research, in: ACM international workshop on Multimedia indexing and information retrieval for healthcare, 2013, pp. 11–14.

[19] A. Ortis, G. M. Farinella, V. DAmico, L. Addesso, G. Torrisi, S. Battiato, Organizing egocentric videos of daily living activities, Pattern Recognition 72 (2017) 207 – 218.

[20] J. Benois-Pineau, M. S. Garcia-Vazquez, L. A. Oropesa Morales, A. A. Ramirez Acosta, Semi-automatic annotation with predicted visual saliency maps for object recognition in wearable video, in: Workshop on Wearable MultiMedia, WearMMe '17, ACM, New York, NY, USA, 2017, pp. 10–14.

[21] A. Fathi, Y. Li, J. M. Rehg, Learning to recognize daily actions using gaze, in: 12th European Conference on Computer Vision, ECCV'12, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 314–327.

[22] Y. Li, Z. Ye, J. M. Rehg, Delving into egocentric actions, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 287–295.

[23] M. Ma, H. Fan, K. M. Kitani, Going deeper into first-person activity recognition, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1894–1903.

[24] I. González-Díaz, J. Benois-Pineau, J. Domenger, A. de Rugy, Perceptually-guided understanding of egocentric video content: Recognition of objects to grasp, in: ACM International Conference on Multimedia Retrieval, ICMR, 2018, pp. 434–441.

[25] D. Pathak, P. Krähenbühl, T. Darrell, Constrained convolutional neural networks for weakly supervised segmentation, in: International Conference on Computer Vision, 2015, pp. 1796–1804.

[26] P. H. O. Pinheiro, R. Collobert, From image-level to pixel-level labeling with convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015, 2015, pp. 1713–1721.

[27] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Is object localization for free? - weakly-supervised learning with convolutional neural networks, 2015, pp. 685–694.

[28] A. Diba, V. Sharma, A. Pazandeh, H. Pirsiavash, L. Van Gool, Weakly supervised cascaded convolutional networks, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5131–5139.

[29] A. Kolesnikov, C. H. Lampert, Seed, expand and constrain: Three principles for weakly-supervised image segmentation, in: European Conference on Computer Vision (ECCV), Springer, 2016, pp. 695–711.

[30] T. Durand, N. Thome, M. Cord, Weldon: Weakly supervised learning of deep convolutional neural networks, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 4743–4752.

[31] T. Durand, T. Mordan, N. Thome, M. Cord, WILDCAT: Weakly Supervised Learning of Deep ConvNets for Image Classification, Pointwise Localization and Segmentation, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5957–5966.

[32] G. Papandreou, L.-C. Chen, K. P. Murphy, A. L. Yuille, Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation, in: IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1742–1750.

[33] H. Bilen, M. Pedersoli, T. Tuytelaars, Weakly supervised object detection with posterior regularization, in: British Machine Vision Conference (BMVC), 2014, p. 1.

[34] H. Bilen, M. Pedersoli, T. Tuytelaars, Weakly supervised object detection with convex clustering., in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2015, pp. 1081–1089.

[35] H. Bilen, A. Vedaldi, Weakly supervised deep detection networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2846–2854.

[36] P. O. Pinheiro, R. Collobert, P. Dollár, Learning to segment object candidates, in: International Conference on Neural Information Processing Systems, NIPS'15, MIT Press, Cambridge, MA, USA, 2015, pp. 1990–1998.

[37] D. P. Papadopoulos, A. D. F. Clarke, F. Keller, V. Ferrari, Training Object Class Detectors from Eye Tracking Data, Springer International Publishing, Cham, 2014, pp. 361–376.

[38] X. Wang, N. Thome, M. Cord, Gaze latent support vector machine for image classification improved by weakly supervised region selection, Pattern Recognition 72 (2017) 59 – 71.

[39] A. Fathi, X. Ren, J. M. Rehg, Learning to recognize objects in egocentric activities, in: CVPR 2011, 2011, pp. 3281–3288.

[40] I. González-Díaz, V. Buso, J. Benois-Pineau, Perceptual modeling in the problem of active object recognition in visual scenes, Pattern Recognition 56 (2016) 129 – 141.

[41] H. Yang, C. Yuan, B. Li, Y. Du, J. Xing, W. Hu, S. J. Maybank, Asymmetric 3d convolutional neural networks for action recognition, Pattern Recognition 85 (2018) 1 – 12.

[42] S. Venugopalan, M. Rohrbach, J. Donahue, R. J. Mooney, T. Darrell, K. Saenko, Sequence to sequence – video to text, in: International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, pp. 4534–4542.

[43] J. Donahue, L. A. Hendricks, M. Rohrbach, S. Venugopalan, S. Guadarrama, K. Saenko, T. Darrell, Long-term recurrent convolutional networks for visual recognition and description, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2017) 677–691.

[44] K. Soomro, H. Idrees, M. Shah, Online localization and prediction of actions and interactions, IEEE Transactions on Pattern Analysis and Machine Intelligence (2018) in press.

[45] C. Fermüller, F. Wang, Y. Yang, K. Zampogiannis, Y. Zhang, F. Barranco, M. Pfeiffer, Prediction of manipulation actions, International Journal of Computer Vision 126 (2018) 358–374.

[46] M. S. Ryoo, Human activity prediction: Early recognition of ongoing activities from streaming videos, in: 2011 International Conference on Computer Vision, 2011, pp. 1036–1043.

[47] V. Bloom, V. Argyriou, D. Makris, Linear latent low dimensional space for online early action recognition and prediction, Pattern Recognition 72 (2017) 532 – 547.

[48] S. Ma, L. Sigal, S. Sclaroff, Learning activity progression in lstms for activity detection and early detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1942–1950.

[49] M. S. A. Akbarian, F. Saleh, M. Salzmann, B. Fernando, L. Petersson, L. Andersson, Encouraging lstms to anticipate actions very early, in: IEEE International Conference on Computer Vision, ICCV, 2017, pp. 280–289.

[50] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (surf), Comput. Vis. Image Underst. 110 (2008) 346–359.

[51] R. I. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, second ed., Cambridge University Press, 2004. ISBN: 0521540518.

[52] M. A. Fischler, R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (1981) 381–395.

[53] D. W. Scott, Multivariate density estimation: theory, practice, and visualization, second edition ed., John Wiley & Sons, Inc, 2015. ISBN: 978-0-471-54770-9.

[54] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in: International Conference on Neural Information Processing Systems, NIPS'15, MIT Press, Cambridge, MA, USA, 2015, pp. 91–99.

[55] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788.

[56] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, 2016, pp. 770–778.

[57] R. Girshick, Fast r-cnn, in: IEEE International Conference on Computer Vision, ICCV '15, IEEE Computer Society, Washington, DC, USA, 2015, pp. 1440–1448.

[58] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (1997) 1735–1780.

[59] F. A. Gers, J. Schmidhuber, F. Cummins, Learning to forget: Continual prediction with lstm, Neural Computation 12 (2000) 2451–2471.

[60] A. Graves, A. Mohamed, G. E. Hinton, Speech recognition with deep recurrent neural networks, in: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013, 2013, pp. 6645–6649.