Extended Gaze Following: Detecting Objects in Videos Beyond the Camera Field of View

Benoit Massé, Stéphane Lathuilière, Pablo Mesejo, Radu Horaud

▶ To cite this version:

HAL Id: hal-02054236
https://inria.hal.science/hal-02054236

Submitted on 1 Mar 2019

Extended Gaze Following:
Detecting Objects in Videos Beyond the Camera Field of View

Benoit Massé[1], Stéphane Lathuilière[1,2], Pablo Mesejo[1,3] and Radu Horaud[1]

[1] Inria & Univ. Grenoble Alpes, France, [2] University of Trento, Italy, [3] University of Granada, Spain

Abstract— In this paper we address the problems of detecting objects of interest in a video and of estimating their locations, solely from the gaze directions of people present in the video. Objects can be indistinctly located inside or outside the camera field of view. We refer to this problem as *extended gaze following*. The contributions of the paper are the followings. First, we propose a novel spatial representation of the gaze directions adopting a top-view perspective. Second, we develop several convolutional encoder/decoder networks to predict object locations and compare them with heuristics and with classical learning-based approaches. Third, in order to train the proposed models, we generate a very large number of synthetic scenarios employing a probabilistic formulation. Finally, our methodology is empirically validated using a publicly available dataset.

I. INTRODUCTION

Gaze following is the ability to intuit the region of space that an observer is looking at. Humans learn this skill during infancy [1], and use it very frequently in many social activities [2]. An accurate estimation of where one or several persons look has an enormous potential in order to determine which are the objects of interest in a scene, predict the actions and movements of the participants and, in general terms, advance towards a better visual scene understanding. It has applications in various fields such as human-robot interaction [3], [4], [5], or action recognition [6]. However, automatically estimating the visual region of attention remains an open challenge, particularly when the gaze target is not visible within the field of view.

This paper addresses the detection of visual regions of attention, which are expected to contain objects of interest. People in a video generally either look at other people or at an object of interest. Such an object can be indistinctly located inside or outside the current image. In the standard *gaze-following* problem, addressed *e.g.* in [7], both the observer and the targeted object are within the same image. An example is provided on Fig. 1(a). This is related to – but significantly different from – estimating the *visual focus of attention i.e.* whom or what a person is looking at [8]. In this case, object locations are known, but potentially non-visible (occluded or outside the field of view, see Fig. 1(b)). However, in a general setting, an object may not be visible within the image, and its location is most probably unknown. All the more in a social interaction, an object is not "of interest" until people actually start paying attention to it. In

this paper, we deal with *extended gaze following* in videos, see Fig. 1(c), meaning that we tackle the more general problem of predicting the location of objects of interest whose number and locations are not known a priori, and that are not necessarily visible.

Our method takes as input a video sequence containing a group of people, and outputs a set of estimated locations for the objects of interest. This work makes the assumption that objects do not move across the video sequence. As in [8], [9], we propose to use the head orientation as a strong cue for the gaze direction. The pipeline is illustrated in Fig. 2.

The contribution of this paper is threefold. First, we propose a novel formalism for embedding the spatial representation of directions of interest and regions of attention. They are modeled as a top-view heat-map, *i.e.* a discrete grid of spatial regions from a top-view perspective. Contrary to the majority of previous work, this formalism is not limited to representing locations within the field of view. Second, we propose several convolutional encoder/decoder neural architectures that learn to predict object locations from head poses in our proposed embedding, and we compare them with several baselines inspired from earlier work. Third, since a large amount of data are required to train a deep neural network, we propose an algorithm based on a generative probabilistic framework that can sample an unlimited number of synthetic conversational scenarios, involving people and objects of interest. The method has been tested both on synthetic data and on a publicly available dataset.

The remainder of this paper is organized as follows. The state of the art is presented in Section II. Then, the details of the proposed heat-map representations and neural network architectures are respectively given in Sections III-A and III-B. The synthetic data generation process is described in Section IV. The Section V is dedicated to experimental results, both on synthetic and real data. To conclude, Section VI discuss the perspectives and limitations of this work.

II. RELATED WORK

Gaze following, or more generally any problem based on the visual attention of a person within an image, is intrinsically based on estimating the gaze direction. In practice, estimating the gaze direction is a complicated problem that still requires to compromise between being precise and non-invasive. When precision is crucial, a head-mounted system, *e.g.* [10], can provide very accurate gaze direction. However,

(a) Gaze following [7] (b) Visual Focus of Attention [8] (c) Extended Gaze following
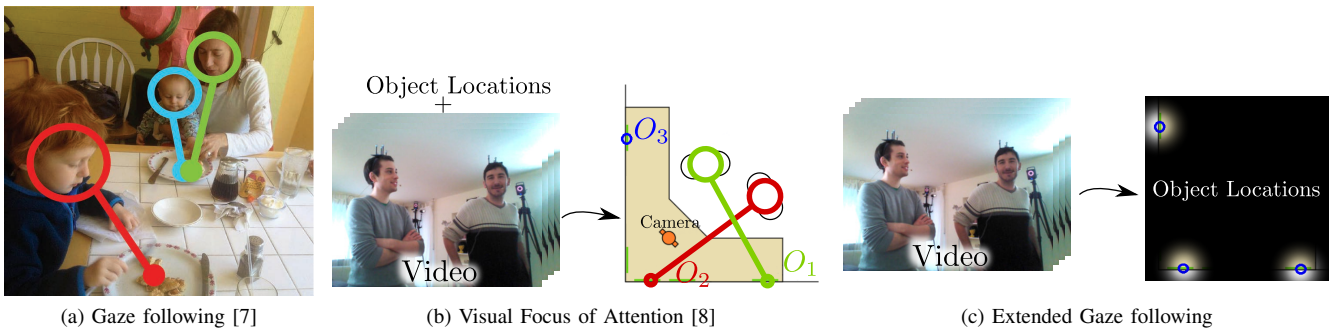
Fig. 1: A comparison of gaze-related computer vision problems. In the standard formulation of gaze following (a), the problem consists in localizing the objects that people are likely to be looking at (and both observer and objects are visible in the input image). Visual focus of attention estimation (b) consists in associating which person is looking at what object at a certain moment (considering that the objects locations are known). In extended gaze following (or visual regions of attention detection) (c), we aim at localizing objects of interest even if they are not visible in the video image.

it cannot be used in a natural scenario since it requires a specific setup. Since the head-mounted system is visible to all participants, it may bias what would be the nature of social behaviour. Another issue is that the head-mounted system can hardly be used to annotate training data since the system would appear in the images recorded by external cameras and, therefore, real environment images that are recorded without head-mounted system would differ from the training set images [11]. On the other hand, estimating gaze direction from remote camera images is a difficult task, with non-frontal faces, or eyelid occlusions [12]. Moreover, since it is difficult to obtain gaze annotations in scenarios where people can move freely, most learning-based methods are trained on extremely simplified setups. For instance, in [13] and [14], subjects were asked to fixate a region on the screen of a camera-equipped device. Alternatively, in unconstrained settings, the head pose is highly correlated with the gaze direction, and the former can be used as an approximation for the latter [9], [8].

Finding objects of interest generally requires to analyze the visual field of view and look for highly contrasting regions. Indeed, an object or a person is likely to look different from the background, thus highly contrasting regions have higher chance of containing something interesting. This approach, similar to the human brain pipeline [15], is known in computer vision as *saliency* [16], [17], [18], [19], where a salient region is one that attracts the visual attention of an observer. In the context of gaze following, the goal is to find regions that are salient, *i.e.* that attracts gaze, from another point of view. However, a salient region is most likely salient from most points of view. Based on this remark, [7] combines a saliency model with a gaze direction model to find salient objects at the intersection of the image and the person's field of view. The attention predictor in [6] also uses both saliency and gaze. By combining multiple gaze directions, [20] estimates shared attention of multiple people, but still within the image. In [21], the authors further investigate this problem based on the idea that the gaze target of a person inside a video may be visible in another video frame. Their

method still relies on a saliency model. Recently, [22] uses a similar combination of gaze and saliency but is also able to predict whether the object of attention lies within the image or not. Finally, [3], [4] merge the problems of saliency and gaze following in the context of human-robot interaction. Indeed, the robot is both an active member of the scenario, and an observer behind the camera. Both papers are based on saliency and gaze direction, as well as additional data such as pointing gesture and speech. However, all works based on saliency require that the object of interest lies within the field of view. By contrast, we wish to be able to locate out-of-view objects; therefore, we cannot rely on this category of methods.

Apart from saliency-based gaze following, a few other methods have been published, addressing the gaze-following problem in the 3D space instead of the 2D image plane. [23] proposes to estimate 3D regions of attention using only the location of people. They model social group structures that constrain the set of candidate locations. In this framework, they learn to locate regions of attention independently of visual saliency. Their method only needs people locations and can work in adversarial scenarios, using only spatial data from first person cameras. However, it fails when some people are undetected and the group structures are wrongly estimated, or when a person is isolated and should not be integrated into a group structure. By contrast, both [24] and [25] independently propose to use 3D intersection of gazes in a probabilistic framework to estimate locations of objects of interest, possibly outside the camera field of view. The methods achieve good levels of performance – even though [24] lacks quantitative evaluation. In both cases, no training data have been used. Each method is designed with strong geometric assumptions so that location inference can be performed without any prior learning phase. At the time this article was written, the data on which the methods have been tested were not released yet for comparison.

In this paper, we combine a learning-based model with a geometric formulation to address the gaze-following prob-
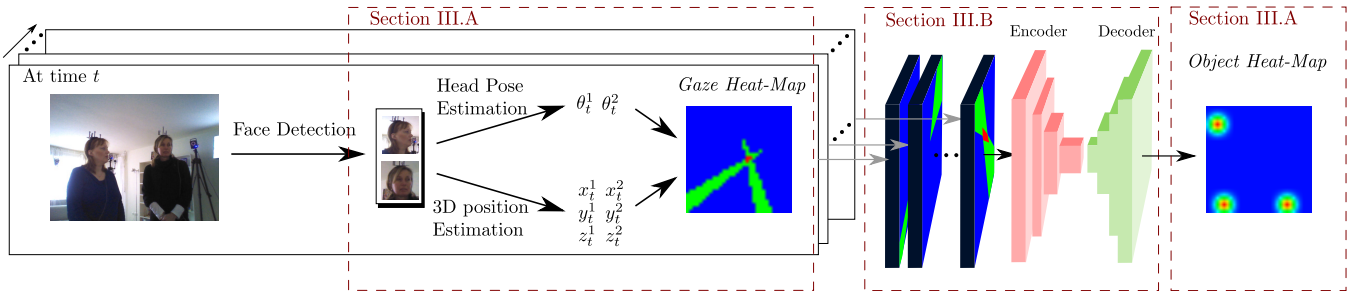
Fig. 2: Outline of the proposed model. For every frame and detected face, orientation and 3D location are estimated, and both sources of information are combined to obtain a top-view representation of the scene encoded in a heat-map. The sequence of heat-maps is then given to a neural network with an encoder/decoder architecture. The network outputs a heat-map that predicts the position of the objects of interest in the top-view domain.

lem, without the restriction of being limited to the image plane. Only very few works exist in this direction [23], [24], [25], and all employ strong social or geometric assumptions.

III. DEEP LEARNING FOR EXTENDED GAZE FOLLOWING

We note N_t the number of persons at time $t \in \{1 \dots T\}$. For each person, we suppose that we can estimate its corresponding 3D head location $[x_t^n, y_t^n, z_t^n]^\top$, and head orientation $[\phi_t^n, \theta_t^n]^\top$ for person $n \in \{1 \dots N_t\}$ in a common scene-centered coordinate frame. However, we additionally choose to drop the z-coordinate (the height) and the head tilt angle as in [24], projecting every object and every person in the same horizontal plane. As we will see later, this simplification drastically reduces the complexity of the model while still representing plausible scenarios. In addition, the tilt angle is commonly the one estimated with the largest mean absolute error [26]. In the remaining of the paper, the term *position* refers to 2D coordinates $\mathbf{x}_t^n = [x_t^n, y_t^n]^\top$ in the horizontal plane (top-view perspective), and *head orientation* refers to the head pan angle ϕ_t^n.

As mentioned before, we decided to use heat-map embeddings. The reasons for this are multiple. First, the exact number of people and objects is not known a priori and may vary within and between video sequences. Heat-map structures are independent of the number of participants (people and objects of interest). Additionally, the problem addressed is fundamentally geometric, and heat-maps intrinsically encode the geometry of the scene. Moreover, convolutional neural networks are able to efficiently extract this structured information in order to obtain a descriptive input representation. A drawback of the heat-map representation is the difficulty to predict an object outside the modeled area. Nevertheless, for indoor scenarios, the area containing the objects is bounded. It is then possible to adapt the heat-map size for the current setup and train the model using scaled simulated scenarios (see Section IV). For all these reasons, we employ heat-map embeddings to model the geometry of the scene.

A. Heat-Map Representation

We propose several heat-map representations of the scene from a top-view perspective. The scene is discretized into a 2D grid of dimension $S_U \times S_V$. Each position in the scene $\mathbf{x} = (x, y)$ is associated to a grid cell $\mathbf{p} = (u, v) \in \{1 \dots S_U\} \times \{1 \dots S_V\}$. As stated previously, $\mathbf{x}$ is bounded in both dimensions: $x \in [x_{min}, x_{max}]$ and $y \in [y_{min}, y_{max}]$. With these notations, $\mathbf{p} = (u, v)$ is obtained from $\mathbf{x}$ as

$$\begin{cases} u &= \lceil S_U \times \frac{x - x_{min}}{x_{max} - x_{min}} \rceil \\ v &= \lceil S_V \times \frac{y - y_{min}}{y_{max} - y_{min}} \rceil \end{cases} \quad (1)$$

where $\lceil \cdot \rceil$ is the *ceiling* function. The grid cell associated to $\mathbf{x}_t^n = (x_t^n, y_t^n)$, the position of a person n at time t, is $\mathbf{p}_t^n$.

In this formalism, a heat-map $\boldsymbol{\Lambda}$ is a 2D map of $S_U \times S_V$ elements that attaches to each cell $\mathbf{p}$ of the grid a value $\boldsymbol{\Lambda}(\mathbf{p})$ between 0 and 1. The meaning of this value depends on what the heat-map represents. In this paper, there are two different categories of heat-map. First, a *gaze heat-map* $\boldsymbol{\Gamma}$ is an embedding for head pose information. A value close to one indicates a region of space consistently situated in front of someone's head. Second, an *object heat-map* $\boldsymbol{\Omega}$ embeds the likelihood for each region to contain an object of interest.

a) Gaze heat-map representation $\boldsymbol{\Gamma}$: Motivated by the use of cones for modeling the dependency between head pose and gaze [28], we compute a heat-map $\boldsymbol{\Gamma}_t^n \in [0,1]^{S_U \times S_V}$ for each person $n \in \{1 \dots N_t\}$ by considering a cone whose axis is the direction spanned by the head pan angle ϕ_t^n. Formally, the value of $\boldsymbol{\Gamma}_t^n$ at any grid cell $\mathbf{p}$ is given by:

$$\boldsymbol{\Gamma}_t^n(\mathbf{p}) = \begin{cases} 1 & \text{if } |\phi(\mathbf{p}) - \phi_t^n| < \epsilon \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where $\phi(\mathbf{p})$ is the angle corresponding to the direction of vector $\overrightarrow{\mathbf{p}_t^n \mathbf{p}}$. The parameter ϵ controls the aperture of the cone. We obtain the *gaze heat-map* illustrated in Fig. 3(b), 3(e) and 3(h):

$$\boldsymbol{\Gamma}_t = \frac{1}{N_t} \sum_{n=1}^{N_t} \boldsymbol{\Gamma}_t^n. \quad (3)$$

(a) Camera Image at $t = 10$

(b) *Gaze heat-map* Γ_{10}

(c) Mean *gaze heat-map* Γ

(d) Camera Image at $t = 30$

(e) *Gaze heat-map* Γ_{30}

(f) Object positions

(g) Camera Image at $t = 80$

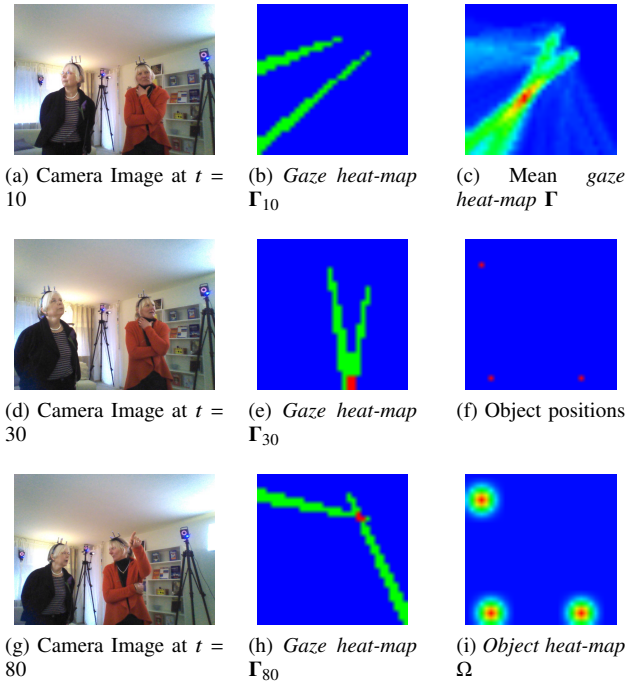(h) *Gaze heat-map* Γ_{80}

(i) *Object heat-map* Ω

Fig. 3: Illustration of the heat-map representations using a sequence extracted from the *Vernissage* dataset [27]. The camera is located close to the bottom left corner of the *gaze heat-maps*. Heat-map colors range from blue to red to indicate number from 0 to 1. (a), (d), (g): camera images. (b), (e), (h): corresponding *gaze heat-maps*. Cone origins in the *gaze heat-maps* indicate people positions; cone axes represent head orientations. (c): mean *Gaze heat-maps* over the sequence. The object ground truth is represented in the heat-map coordinate frame (f). This provides the ground truth *Object heat-map* (i) used for training and *MSE* evaluation.

It is sometimes useful to aggregate the *gaze heat-maps* through time into a mean *gaze heat-map* (see Fig. 3(c)) to have an compact representation of the scenario:

$$\Gamma = \frac{1}{T} \sum_{t=1}^{T} \Gamma_t. \tag{4}$$

b) Object heat-map Ω: Considering a scenario with M objects (*e.g.* Fig. 3(f)), we compute a heat-map $\Omega \in [0,1]^{S_U \times S_V}$ (Fig. 3(i)) whose value at grid cell $\mathbf{p}$ is given by:

$$\Omega(\mathbf{p}) = \max_{1 \le m \le M} \exp\left(-\frac{||\mathbf{p} - \mathbf{p}_{obj}^m||_2^2}{2\sigma_\Omega^2}\right) \tag{5}$$

where $\mathbf{p}_{obj}^m$ is the grid cell corresponding to the scene position of the mth object. The variance σ_Ω controls the spread of the peaks. As objects do not move, Ω remains constant during a scenario.

Now, let us suppose we have been able to obtain an estimate $\hat{\Omega}$ of Ω from $\Gamma_1 \ldots \Gamma_T$. Finally, to obtain an actual list of object positions, we extract the local maxima from $\hat{\Omega}$ and discard local maxima that are too low compared to the

global maximum. More precisely, given a candidate position $\mathbf{p}_C$, a neighborhood of this position $\mathcal{N}(\mathbf{p}_C)$ and a shrinking function $\alpha(\cdot)$ such that $\alpha(x) \le x$, we consider that $\mathbf{p}_C$ contains an object if

$$\mathbf{p}_C = \underset{\mathbf{p} \in \mathcal{N}(\mathbf{p}_C)}{\operatorname{argmax}} \hat{\Omega}(\mathbf{p}) \quad \text{and} \quad \hat{\Omega}(\mathbf{p}_C) \ge \alpha\left(\max_{\mathbf{p}} \hat{\Omega}(\mathbf{p})\right). \tag{6}$$

The section III-B below is dedicated to propose a neural network that learns to predict an estimate $\hat{\Omega}$ of the *object heat-map* from the set of *gaze heat-maps* $\Gamma_1 \ldots \Gamma_T$.

B. Object heat-map inference

Now, we address the problem of estimating $\hat{\Omega}$, on which the local maxima detection algorithm can be run. We propose several baselines with justification for their relevance. Then, we present our architectures based on convolutional encoder/decoder.

c) Heuristics without learning: First, we propose two heuristics with no training. The local maxima detection is performed directly on a combination of gaze heat-maps. Indeed, the regions that are activated (close to one) in multiple gaze heat-maps are consistently in front of someone's head and have a high chance of containing an object. Previous works [24], [28] already used geometric features based on cone intersections. The heuristics are as follow.

- *Cone*: The local maxima extraction is performed directly on the *mean gaze heat-map* $\Gamma = \frac{1}{T} \sum_{t=1}^{T} \Gamma_t$.
- *Intersect*: We define a *gaze intersection heat-map* Γ_t^{inter} per time frame, by setting regions to one only if they are at the intersection of multiple cones. More formally,

$$\Gamma_t^{inter}(\mathbf{p}) = \begin{cases} 1 & \text{if } \sum_{n=1}^{N_t} \Gamma_t^n(\mathbf{p}) \ge 2 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

The local maxima extraction is performed on $\Gamma^{inter} = \frac{1}{T} \sum_{t=1}^{T} \Gamma_t^{inter}$.

d) Learning-based Baselines: We define some simple regression models. They learn a regression from the *mean gaze heat-map* $\Gamma = \frac{1}{T} \sum_{t=1}^{T} \Gamma_t$ to the *Object heat-map* $\hat{\Omega}$. These models consider the input and output as flattened vectors of $S_U \times S_V$ components.

- *Linear Reg.*: We learn a linear regression model from Γ to $\hat{\Omega}$. Interestingly, the output of a linear regression is not constrained to lie between 0 and 1, contrary to the definition of Ω. The local maxima extraction is performed after $\hat{\Omega}$ has been rescaled in $[0,1]$.
- *d-FC*: The regression is performed on Γ by a network composed of $d \in \{1,3\}$ fully connected hidden layers of $S_U \times S_V$ units, with ReLU activations. The last hidden layer is fully connected to the output *object heat-map* with sigmoid activations.

e) Encoder/Decoder Architectures: They have been used for many computer vision tasks where the goal is to perform a regression between high dimensional spaces [29], [30]. Such architectures are composed of two sub-networks, where the first reduces the spatial resolution of the input to obtain a compact description of it, and the second alternates between up-sampling and fully-connected layers until recovering a high dimensional output. In our particular problem, we use convolutional layers instead of fully-connected layers to model the spatial connections. Moreover, as the input is a sequence, several encoder architectures can be employed. We propose to use a decoder composed of three successive up-sampling and convolutional layers with 3×3 kernels. The last convolution layer of the decoder employs sigmoid activations. The whole network is trained employing the Mean Squared Error (MSE) loss. We propose the four following architectures that represent a progressively increasing complexity. Graphical representations of the proposed networks are given in the supplementary material[1].

- *Mean-2D-Enc*: This is the simplest model. We use the *mean gaze heat-maps* Γ as in the baselines. It is fed to a standard 2D convolutional encoder composed of three successive convolutional and down-sampling layers.
- *2D-Enc*: In this model, we consider that time plays the role of the color-axis in standard 2D convolutions. $\Gamma_1 \ldots \Gamma_T$ are concatenated along the third dimension to obtain the *sequence gaze heat-map* $\Gamma_{1:T}$. Therefore, the first layer kernels have as dimension $3 \times 3 \times T$ instead of $3 \times 3 \times 1$ like in *Mean-2D-Enc*.
- *3D-Enc*: Inspired by [31], that shows that 3D convolutions are able to extract reliable features from both the spatial and the temporal dimensions, we propose a 3D-Encoder network on $\Gamma_{1:T}$. By performing 3D convolutions, the model can capture orientation changes and people motion in successive frames. The time dimension is reduced, from T to 1 after three convolutional and max-pooling layers, before feeding it to the 2D-Decoder.
- *3D/2D U-Net*: This variant of the *3D-Enc* architecture is inspired from the U-Net architecture [32]. In our specific case, since we have a 3D encoder, we need to squeeze the time dimension. To do so, we combine over time the feature maps of the encoder with max-pooling, before concatenation to the decoder.

IV. SYNTHETIC SCENARIO GENERATION FOR NETWORK TRAINING

A large amount of data is required to train deep networks. Unfortunately, obtaining such a dataset is difficult, since, in practice, we would need to know the true object locations for every sequence. For instance, in the *Vernissage* dataset [27], objects outside the field of view have been annotated employing infrared cameras. This setting is well-suited for our problem but it would be difficult to obtain a sufficiently large and diverse dataset of object locations to train deep networks.

[1]see *https://team.inria.fr/perception/research/extended-gaze-following*

Consequently, Vernissage is used only to test our model and not to train it.

To face this issue, we propose to use synthetically generated data. More precisely, we simulate scenarios involving people and objects, and generate their corresponding input sequences and associated true object locations. We define a probabilistic model that relates the object 2D positions and the head poses, and generate samples according to the underlying distribution. We now aim at generating a scenario of length T involving a constant number N of people with respective positions $\mathbf{x}_{1:T}^n$ and orientations $\phi_{1:T}^n$, given $1 < n < N$; and M objects located at the positions $\mathbf{x}_{obj}^m, 1 < m < M$. To this aim, we define the joint distribution $P(\phi_{1:T}^{1:N}, \mathbf{x}_{1:T}^{1:N}, \mathbf{x}_{obj}^{1:M})$ considering the following factorization:

$$\underbrace{P(\phi_{1:T}^{1:N}|\mathbf{x}_{1:T}^{1:N}, \mathbf{x}_{obj}^{1:M})}_{\substack{\text{Head orientation} \\ \text{distribution}}} \times \underbrace{P(\mathbf{x}_{1:T}^{1:N}|\mathbf{x}_{obj}^{1:M})}_{\substack{\text{People motion} \\ \text{distribution}}} \times \underbrace{P(\mathbf{x}_{obj}^{1:M})}_{\substack{\text{Object position} \\ \text{distribution}}} \quad (8)$$

The *object position distribution* $P(\mathbf{x}_{obj}^{1:M})$ is based on a uniform distribution within the top-view grid, since we want to have a high variety of settings. However, some settings are even too difficult for a human to distinguish between objects. For this reason, the generator can choose to resample an object under two criteria. First, the closest two objects are from each other, the highest the chance one of them is resampled. Therefore, we impose that objects have a minimal physical size and that two objects cannot be one above the other. Second, objects too far from the heat-map edges also have a high chance of being resampled. In many scenarios, objects of interest tend to be close to the walls, *e.g.* posters, computer screens, paintings in a museum. This tends to reduce the number of ambiguous cases in which several objects are aligned from the point of view of someone.

Importantly, in a human-robot interaction scenario, people may look at the robot, but we want to avoid our model to predict the presence of an object at the robot camera position. Therefore, as the camera position $\mathbf{x}_{camera}$ is known, we propose to add a blank object at the corresponding grid cell $\mathbf{p}_{camera}$ in all sequences. The blank object behave like normal objects – constant position, can be gazed at – but does not appear in the *object heat-map* at training time and thus should be ignored at prediction time. Also, it cannot be resampled while generating the objects.

Concerning the *people motion distribution*, $P(\mathbf{x}_{1:T}^{1:N}|\mathbf{x}_{obj}^{1:M})$, we describe first how the initial positions $\mathbf{x}_1^{1:N}$ are sampled, and then how each $\mathbf{x}_{t+1}^n$ is sampled iteratively from $\mathbf{x}_t^n$. First, the initial positions of people are obtained similarly to object positions. Namely, they are sampled uniformly within the boundaries, and can be resampled when too close to an object, another person, or (contrary to objects) too close to the edges. Concerning the motion, we consider that people can either stay still for a random period of time, or move linearly short distances. In practice, there is a high probability that the person stay still $\mathbf{x}_{t+1}^n = \mathbf{x}_t^n$. Otherwise,
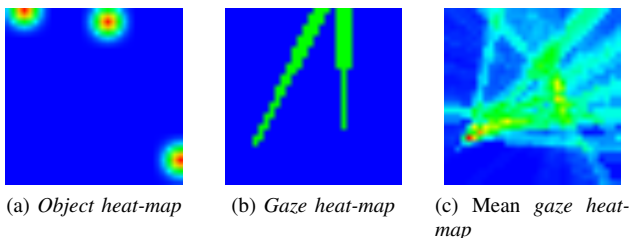
(a) *Object heat-map* (b) *Gaze heat-map* (c) Mean *gaze heat-map*

Fig. 4: Heat-maps from a synthetic scenario generated randomly, with 2 people ($N = 2$) and 3 objects ($M = 3$). (a): the ground truth *Object heat-map* Ω used for training or evaluation. (b): a *Gaze heat-map* randomly chosen among the sequence. (c): the mean *gaze heat-map* over the sequence.

$\mathbf{x}_{t+\tau}^n$ is sampled from a normal distribution centered on $\mathbf{x}_t^n$, and possibly resampled as long as $\mathbf{x}_{t+\tau}^n$ is outside the boundaries or too close to another target. In the latter case, $\mathbf{x}_{t+1}^n \ldots \mathbf{x}_{t+\tau-1}^n$ are linearly interpolated.

Finally, for the *head orientation distribution*, we define a probabilistic model inspired by [8], where the authors propose a method to estimate the visual focus of attention of multiple people by applying Bayesian inference on a generative model. In this probabilistic model, the head orientation dynamics are explained by some latent variables, *e.g.* gaze direction. For more details, see [8]. In our case, we propose to sample the latent variables over time, then sample the head orientation ϕ_t^n given the latent variables. This model is well-suited for our sampling task since multiple situations may occur, *e.g.* mutual gaze or joint attention, that are treated differently by their temporal formulation. Moreover, it takes into account the discrepancy between head pose and gaze direction, and the network can learn this difference because it is modeled at training time.

The Fig. 4 represents a synthetic scenario generated using this process. In practice, a wide variety of scenarios can be obtained with this approach. For instance, there is no limit to the number of people and/or objects that could be generated in one scenario, except the plausibility of such a scenario with respect to the physical space.

V. EXPERIMENTS

Experiments have been performed both on *synthetic* data, generated online as described in section IV, and on the *Vernissage* dataset [27] as described below. Note that, we do not use the datasets employed in [24] and [25] since they are not publicly available

f) The Vernissage *Dataset:* It is composed of ten recordings lasting approximately ten minutes each. Each sequence contains two people interacting with a Nao robot and discussing about three wall paintings ($M = 3$). The robot plays the role of an art guide, describing the paintings and asking questions to the people in front of it. The scene was recorded at 25 frames per second (fps) with an RGB camera embedded into the robot head, and with a VICON motion

capture system consisting of a network of infrared cameras providing accurate position and head pose estimations of the people, the objects and the robot. We use the OpenCV version of [33] for face detection and [34] to track the faces over time. The head poses are estimated by employing [26]. The 3D head positions, are estimated using the face center and the bounding-box size, which provides a rough estimate of the depth. The position of the robot itself and the orientation of its head are also known. Finally, the object locations are annotated along with the visual focus of attention of the participants over time. Images extracted from Nao camera during various recordings are displayed in Fig. 1(b), 1(c), 2, 3(a), 3(d), 3(g).

g) Implementation details: The heat-map dimensions are set to $S_X = S_Y = 32$, to represent a room of size $3m \times 3m$. The cone aperture ϵ is set to $2°$. We fixed the input sequence size to $T = 200$ time steps. On the *Vernissage* dataset, the videos are subsampled to 5 fps, then the duration of a sequence is $40s$ and we can extract several sequences from each video sequence. By using a sliding window and 50% overlap, we extract a total of 224 sequences. We use the visual focus of attention annotations to obtain the true objects of interest for each sequence. Consequently, the number of objects can vary from 1 to 3 in the test sequences. We employ the adam optimizer [35] for 10 epochs. For all neural network architectures employed in the experiments, the batch size is set to 32. In all cases, we perform the local maxima extraction method described in (6) after estimating $\hat{\Omega}$ to obtain the list of object positions. The neighborhood $\mathcal{N}(\cdot)$ from (6) is defined as a sliding region of 5×5 pixels, and the shrinking function $\alpha : x \mapsto \ln(1+x)$. In all our experiments, we report *Precision* and *Recall*, and these two metrics are combined to obtain the *f1-score*. *Precision* measures the percentage of detected objects that are true objects. *Recall* measures the percentage of true objects correctly detected. In order to compute these metrics, we employ a Hungarian algorithm that matches the detections with the real objects positions based on their respective distances. Importantly, the detection is considered as a success if the distance between the estimated and annotated distances is lower than 50cm in the real-world space. For all learning-based approaches, we also report the MSE between the predicted and true *object heat-maps*. Since Heuristic methods do not intend to predict the *object heat-maps*, the MSE is not reported for them.

h) Results and Discussion: In Table I, we report the results obtained employing all methods described on both *synthetic* and real data.

It has to be noted that many different recurrent architectures have been considered, either alone or in conjunction with one of the proposed convolutional Encoder/Decoder architectures *e.g.* adapted from the convolutional LSTM [36]. All of them converged to networks predicting always the same (or almost the same) *object heat-map*. We believe that, in this formulation, the ability to combine information from distant time frame is important, and this is difficult to achieve with RNN (or LSTM) processing data sequentially [37].

TABLE I: Results obtained on data from the proposed synthetic generator and on the *Vernissage* dataset [27]. MSE values reported were multiplied by 10^2 to facilitate reading. *Precision*, *recall* and *f1-score* represent percentages. For learning-based approaches, we report the mean and standard deviation over five runs. Results on the [25] dataset are reported for comparison.

Dataset	Synthetic			
Method	*MSE*	*Precision*	*Recall*	*f1-score*
Cone	-	18.8	53.9	27.8
Intersect	-	21.1	35.0	26.3
Linear Reg.	1.25 ± 0.02	50.5 ± 2.2	76.9 ± 1.0	60.9 ± 1.8
1-FC	1.06 ± 0.03	64.9 ± 1.6	61.5 ± 1.5	63.1 ± 1.1
3-FC	1.05 ± 0.01	65.9 ± 0.6	59.9 ± 2.2	62.8 ± 1.2
Mean-2D-Enc	1.00 ± 0.03	74.5 ± 2.4	59.5 ± 1.7	66.1 ± 1.3
2D-Enc	0.98 ± 0.02	76.8 ± 2.2	62.2 ± 1.5	68.7 ± 1.7
3D-Enc	0.85 ± 0.06	88.2 ± 3.9	71.4 ± 2.1	78.9 ± 2.4
3D/2D U-Net	**0.75 ± 0.01**	**89.0 ± 1.2**	**78.0 ± 0.6**	**83.2 ± 0.8**
Dataset	Vernissage			
Method	*MSE*	*Precision*	*Recall*	*f1-score*
Cone	-	20.7	35.8	26.2
Intersect	-	34.9	27.2	30.6
Linear Reg.	1.48 ± 0.04	37.0 ± 4.9	**53.7 ± 5.0**	43.7 ± 4.6
1-FC	1.49 ± 0.02	29.9 ± 3.2	35.2 ± 2.5	32.3 ± 2.8
3-FC	1.49 ± 0.02	28.0 ± 3.5	29.9 ± 1.5	28.8 ± 2.4
Mean-2D-Enc	**1.37 ± 0.02**	**60.1 ± 1.5**	41.1 ± 1.0	**48.8 ± 1.2**
2D-Enc	1.39 ± 0.03	54.9 ± 4.2	40.5 ± 1.6	46.6 ± 2.5
3D-Enc	1.43 ± 0.05	49.9 ± 8.1	37.1 ± 9.0	42.5 ± 8.7
3D/2D U-Net	1.44 ± 0.04	45.1 ± 4.8	38.5 ± 2.2	41.5 ± 3.3
Dataset	Brau et al. [25]			
Method	*MSE*	*Precision*	*Recall*	*f1-score*
Brau et al. [25]	-	59.0	48.0	*52.9*

(a) $\hat{\Omega}$ - *Mean-2D-Enc* (b) $\hat{\Omega}$ - *3D/2D U-Net* (c) $\hat{\Omega}$ - *Linear Reg.*

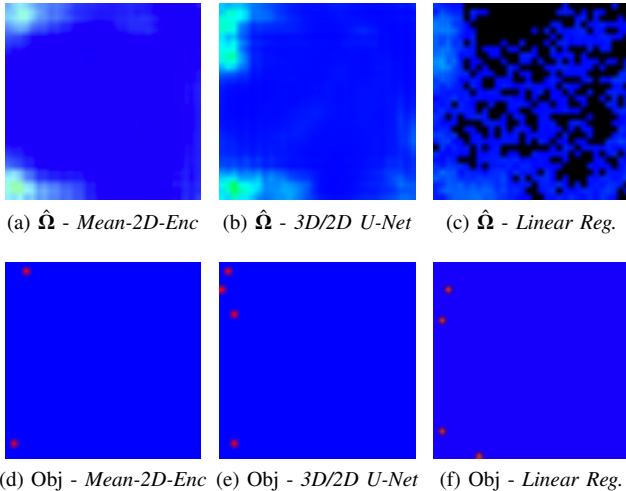(d) Obj - *Mean-2D-Enc* (e) Obj - *3D/2D U-Net* (f) Obj - *Linear Reg.*

Fig. 5: Results of three methods on the *Vernissage* scenario illustrated in Fig. 3. 5(a), 5(b), 5(c): Estimates $\hat{\Omega}$ of the *Vernissage object heat-map* Ω from Fig. 3(i) using three different architectures. 5(d), 5(e), 5(f): Corresponding objects positions, obtained as the highest local maxima from $\hat{\Omega}$. Black pixels in 5(c) indicate negative values.

From the experiments, we observe that learning-based approaches clearly outperform those based on cone intersections inspired from [24]. Indeed, even on the *synthetic* datasets, their *precision* and *recall* do not reach better than 18.8% and 53.9% respectively, whereas a simple linear regression reaches considerably higher scores (50.5% and 76.9% respectively). The same remark stands for the *Vernissage* dataset. Increasing the network complexity by simply adding fully-connected layers does not bring any improvement and even reduce the performance. Then, we observe that all proposed encoder/decoder models clearly outperform other methods by a substantial margin on the *synthetic* dataset. There, we obtain a 22.3% gain in terms of f1-score when employing the *3D/2D U-Net* with respect to the linear regression model. On the *Vernissage* dataset, a 5.1% gain is obtained in terms of *f1-score* when employing the *Mean-2D-Enc* with respect to the linear regression model. These experiments validate the use of the encoder/decoder architecture.

We notice that the performance on the *synthetic* dataset increases with encoder complexity. However, the inverse phenomenon is observed on *Vernissage*, where the best performances are obtained using the simplest encoder architecture (that does not model time). Our guess for this observation is that there is a significant discrepancy between the distribution of *Vernissage* data and the *synthetic* data distribution sampled according to (8). Therefore, more complex models probably tend to over-fit the *synthetic* data distribution, and thus transfer less well on the *Vernissage* dataset. More realistic training data could lead to further improvements. This could be obtained by gathering a dataset of real-life scenarios which could be use either as training data or to improve the quality of the generative model.

The only methods from the literature that we are aware of are [24] and [25]. In both cases, neither the data nor the code have been made available online. Moreover, the papers lack information about parameters or hyperparameters that prevented us to test it. Additionally, [25] explicitly discarded the *Vernissage* dataset in their experiments. Results on their dataset (59% precision and 48% recall) are comparable to ours on *Vernissage*. Note that, [25] employed a larger success threshold (1.0m in the real-world space for 50cm in our case) and consequently would obtain lower scores according to our evaluation protocol. We wish to test our method on their dataset in the future. We do not compare to [24] since they did not report any quantitative results on location estimation.

In Fig. 5, the predicted *gaze heat-maps* $\hat{\Omega}$ for several learning-based approaches applied on the scenario from Fig. 3 are displayed. The architectures *Mean-2D-Enc* and *Linear Reg.* use the average *gaze heat-map* $\frac{1}{T} \sum_{t=1}^{T} \Gamma_t$ as input, whereas *3D/2D U-Net* takes the whole concatenated sequence $\Gamma_{1:T}$. All three approaches are approximately able to predict the positions of two objects of interest. The third object is probably not targeted enough during the sequence to be found. The black pixels in the *Linear Regression* indicate negative values. All other approaches end with a

sigmoid activation so each pixel value is homogeneous to a probability. The lower number of falsely proposed object positions for the *Mean-2D-Enc* is consistent with the higher mean precision reported. For comparison, experiments on the *synthetic* scenario from Fig. 4 are available in the supplementary materials.

VI. CONCLUSIONS

In this paper, we define the problem of *extended gaze following* as finding the locations of objects of interest solely from the gaze direction of visible people. Importantly, this allows for finding objects either inside or outside the camera field-of-view. In this context, we propose a novel spatial representation for head poses (approximating gaze direction) and object locations. We present a framework that takes advantage of convolutional encoder/decoder architectures to learn the spatial relationship between head poses and object locations, and we compare nine different methods on synthetic and real data. We finally conclude that learning-based approaches outperform geometry-based ones while being competitive with the state of the art. We also show that the necessary training examples can be quickly and easily obtained through a synthetic data generation process.

We believe this work open new perspectives for research. In particular, several decisions were taken to obtain an end-to-end method (*e.g.* heat-map representation or elevation coordinate omission), which makes it hardly suitable in some situations. The extended gaze-following problem would benefit greatly from a benchmark of different representations and inference models, and of the influence of each simplifying hypothesis. Moreover, the availability of suitable datasets would ease future research on this topic. In parallel, we wish to use this framework in the future as a tool to improve the decision process of a robotic system in a social context such as [5].

REFERENCES

[1] Dare A Baldwin. Understanding the link between joint attention and language. *Joint attention: Its origins and role in development*, 1995.

[2] Michael Land and Benjamin Tatler. *Looking and acting: Vision and eye movements in natural behaviour*. Oxford University Press, 2009.

[3] Boris Schauerte and Rainer Stiefelhagen. "Look at this!" Learning to guide visual saliency in human-robot interaction. In *IROS*, 2014.

[4] Joris Domhof, Aswin Chandarr, Maja Rudinac, and Pieter Jonker. Multimodal joint visual attention model for natural human-robot interaction in domestic environments. In *IROS*, 2015.

[5] Stéphane Lathuilière, Benoit Massé, Pablo Mesejo, and Radu Horaud. Neural network based reinforcement learning for audio-visual gaze control in human-robot interaction. *Pattern Recognition Letters*, 118:61–71, 2019.

[6] Ping Wei, Yang Liu, Tianmin Shu, Nanning Zheng, and Song-Chun Zhu. Where and why are they looking? jointly inferring human attention and intentions in complex tasks. In *IEEE CVPR*, 2018.

[7] Adria Recasens, Aditya Khosla, Carl Vondrick, and Antonio Torralba. Where are they looking? In *NIPS*, 2015.

[8] Benoit Massé, Silèye Ba, and Radu Horaud. Tracking gaze and visual focus of attention of people involved in social interaction. *IEEE TPAMI*, 2018.

[9] Sankha S Mukherjee and Neil Martin Robertson. Deep head pose: Gaze-direction estimation in multimodal video. *IEEE TMM*, 2015.

[10] Aun Kei Ariana Hong, Jeff Pelz, and Juan Cockburn. Lightweight, low-cost, side-mounted mobile eye tracking system. In *IEEE WNYIPW*, 2012.

[11] Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. Rt-gene: Real-time eye gaze estimation in natural environments. In *ECCV*, 2018.

[12] Dan Witzner Hansen and Qiang Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE TPAMI*, March 2010.

[13] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye tracking for everyone. In *IEEE CVPR*, 2016.

[14] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. In *IEEE CVPR*, 2015.

[15] Stefan Treue. Visual attention: the where, what, how and why of saliency. *Current Opinion in Neurobiology*, 2003.

[16] Laurent Itti and Christof Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2001.

[17] Dmitry Rudoy, Dan B Goldman, Eli Shechtman, and Lihi Zelnik-Manor. Learning video saliency from human gaze using candidate selection. In *IEEE CVPR*, 2013.

[18] Daniel Parks, Ali Borji, and Laurent Itti. Augmented saliency model using automatic 3d head pose detection and learned gaze following in natural scenes. *Vision research*, 2015.

[19] Wenguan Wang and Jianbing Shen. Deep visual attention prediction. *IEEE TIP*, 2018.

[20] Lifeng Fan, Yixin Chen, Ping Wei, Wenguan Wang, and Song-Chun Zhu. Inferring shared attention in social scene videos. In *IEEE CVPR*, 2018.

[21] Adria Recasens, Carl Vondrick, Aditya Khosla, and Antonio Torralba. Following gaze in video. In *CVPR*, 2017.

[22] Eunji Chong, Nataniel Ruiz, Yongxin Wang, Yun Zhang, Agata Rozga, and James M. Rehg. Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of gaze and scene saliency. In *ECCV*, 2018.

[23] Hyun Soo Park and Jianbo Shi. Social saliency prediction. In *IEEE CVPR*, 2015.

[24] Meir Cohen, Ilan Shimshoni, Ehud Rivlin, and Amit Adam. Detecting Mutual Awareness Events. *IEEE TPAMI*, 2012.

[25] Ernesto Brau, Jinyan Guan, Tanya Jeffries, and Kobus Barnard. Multiple-gaze geometry: Inferring novel 3d locations from gazes observed in monocular video. In *ECCV*, 2018.

[26] Stéphane Lathuilière, Rémi Juge, Pablo Mesejo, Rafael Muñoz-Salinas, and Radu Horaud. Deep mixture of linear inverse regressions applied to head-pose estimation. In *IEEE CVPR*, 2017.

[27] Dinesh Babu Jayagopi, Samira Sheiki, David Klotz, Johannes Wienke, Jean-Marc Odobez, Sebastien Wrede, Vasil Khalidov, Laurent Nyugen, Britta Wrede, and Daniel Gatica-Perez. The vernissage corpus: A conversational human-robot-interaction dataset. In *IEEE HRI*, 2013.

[28] Manuel J. Marin-Jimenez, Andrew. Zisserman, Marcin Eichner, and Vittorio. Ferrari. Detecting people looking at each other in videos. *IJCV*, 2014.

[29] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[30] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE TPAMI*, 2017.

[31] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3D convolutional neural networks for human action recognition. *IEEE TPAMI*, 2013.

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*. Springer, 2015.

[33] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE CVPR*, 2001.

[34] Sileye Ba, Xavier Alameda-Pineda, Alessio Xompero, and Radu Horaud. An on-line variational bayesian model for multi-person tracking from cluttered scenes. *CVIU*, 2016.

[35] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014.

[36] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *IEEE CVPR*, 2015.

[37] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *ICML*, 2013.