Evaluating Single-Task and Multi-Task Models for Binary and Continuous Saliency Map Prediction

Alessandro Di Stefano[1]

[1]Department of Computer Science, University of Milan, Milan, Italy.

Contributing authors: alessandro.distefano2@studenti.unimi.it;

Abstract

In many social contexts, an observer's visual attention is influenced by the behavior of people in the scene, such as their gaze direction or speech. Under these circumstances, saliency maps can serve as a useful tool to model and quantify such socially guided attention patterns. This paper evaluates the effectiveness of saliency maps as both an input representation and a target for fixation point prediction. We address two tasks: predicting binary fixation maps and predicting continuous fixation maps. For each task, we compare a single-task learning approach, trained exclusively on the corresponding objective, with a multi-task learning approach that jointly optimizes both tasks. Experimental results highlight the relative strengths of single-task and multi-task models, revealing how different task formulations shape the predictive capacity of fixation models.

1 Introduction

Humans, like other primates, direct their attention toward group members to acquire valuable social information, such as identity, dominance, emotions, and intentions. In social contexts, an observer's visual attention is naturally drawn to socially relevant elements, particularly faces and eyes, which provide critical cues for inferring the intentions, attention, and affective states of others[1]. Biologically salient stimuli, or objects with high "animacy," strongly capture visual attention. Computationally, such attention patterns can be modeled using saliency maps, which estimate the likelihood that observers will fixate on different regions of a scene, thereby providing a framework to capture and predict socially relevant visual cues. Computational models that assign a saliency value to each pixel of an image are generally referred to as saliency models. Over the years, numerous saliency models have been proposed, yet determining which model most accurately predicts human eye fixations remains a challenging problem. One reason is that evaluating a model's predictive ability is itself an open research question, as the choice of evaluation metric depends on how saliency is defined and how ground truth fixation data are represented[2] Another fundamental challenge arises from the discrete nature of human fixation points. In practice, eye-tracking data record exact fixation locations, which are sparse and discrete. Most saliency models, especially those based on deep learning (DL), instead predict continuous maps, where each pixel encodes the probability of visual attention. This discrepancy between discrete ground truth fixations and continuous model predictions can complicate direct optimization and limit precise modeling of human fixations. Despite these challenges, deep learning has significantly advanced saliency prediction in recent years, largely due to the availability of large-scale annotated datasets. Collecting these datasets typically involves presenting images to multiple human observers while tracking their eye movements. Individual fixations are then aggregated and smoothed (e.g. with a Gaussian filter) to produce continuous saliency maps, which most models use as training targets. While this approach facilitates learning and generalization, it also highlights a key limitation: the architectures do not explicitly replicate the discrete fixation generation process, which can complicate precise modeling of fixation locations[3] The goal of this paper is to assess the capabilities of a deep learning (DL) model in generating accurate saliency maps from a set of fixation points. Specifically, we aim to compare the performance of a multi-task model, trained to generate both binary and continuous saliency maps, with that of a single-task model, trained to generate only one specific type of map (either binary or continuous). The rest of this paper is organized as follows. Section 2 formally outlines the task that we seek to address. Section 3 describes the proposed approach. Section 4 introduces the experimental setup, while Section 5 shows the final results.

2 The Addressed Task

This study focuses on **social scenes** extracted from a set of videos containing k faces. For each video, fixation data from $m = 39$ external observers are available, providing a rich dataset of human visual attention patterns in social contexts. The addressed task comprises two main phases: constructing input and output saliency maps, and designing predictive models. For the **input saliency maps**, gaze points are estimated by applying a **gaze estimation model** to frames extracted from the videos. These gaze points are then aggregated to build the corresponding input maps, highlighting regions where observers are likely to focus. For the **output saliency maps**, two types of maps are generated:

1. **Binary maps**: maps representing the N discrete human fixation points. Each pixel is assigned a value of 1 (white) if it corresponds to a fixation location, and 0 (black) otherwise. While this format directly reflects the ground truth fixation targets, its sparsity poses challenges for optimization-based models, which may struggle to learn from such limited signal.

2. **Continuous maps**: these maps are created by placing **Gaussian blobs** centered on each discrete fixation point, producing a continuous distribution of fixation likelihood across the image. Each map is normalized by dividing pixel values by the global maximum, preserving peak intensities while ensuring consistent scaling.

Finally, three predictive models are considered:

- A **binary model**, trained to predict only the binary saliency map.
- A **continuous model**, trained to predict only the continuous saliency map.
- A **multi-task model**, trained to jointly predict both binary and continuous maps.

This framework allows us to systematically compare **single-task** and **multi-task learning** approaches in saliency prediction. An example of the pipeline of the addressed task is illustrated in Fig.1.

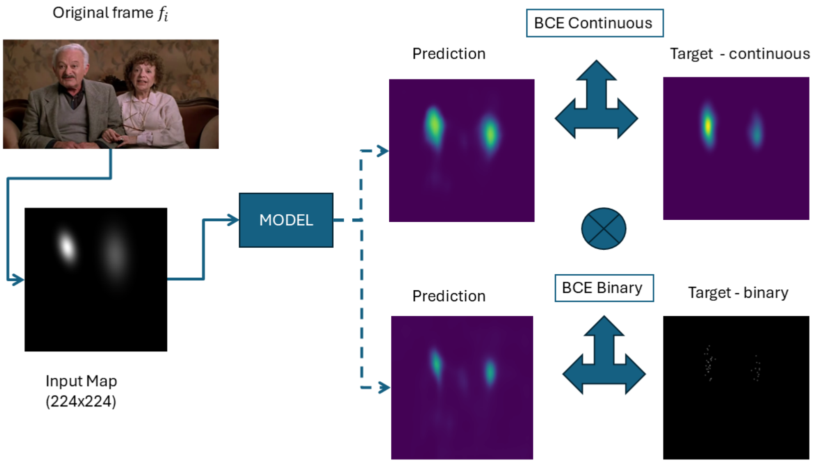

Fig. 1 Schematic representation of the addressed task. Video frames are processed by a gaze estimation model to obtain input saliency maps. Output maps are generated in binary and continuous formats and used to train binary, continuous, and multi-task models.

3 Methodology

This section describes the full pipeline for saliency prediction in social scenes, including the construction of gaze-based input saliency maps and the generation of output fixation maps. We used a pretrained gaze estimation model to infer attention cues and applied a procedure to estimate spatial uncertainty and generate fixation-based saliency representations.

3.1 Gaze-Based Input Map Construction

To generate input saliency maps, we applied the **Gaze360** model[6] to each cropped face region within the video frames. Gaze360 is a deep learning model trained on unconstrained gaze data, capable of estimating 3D gaze direction in spherical coordinates, along with an associated **offset** value that reflects prediction uncertainty. The model outputs the expected gaze direction as a unit vector in spherical space, which we converted to Cartesian coordinates and projected onto the 2D image plane using standard perspective projection equations. Since the gaze vector encodes direction without magnitude, we applied a fixed-length projection calibrated to the size of the cropped face region.

3.2 Uncertainty-Aware Gaussian Blob Generation

To account for gaze uncertainty, we used the offset information provided by Gaze360 model to estimate variability in the predicted gaze direction. Specifically, we assumed that the estimation follows a Gaussian distribution, and used the provided 10th and 90th percentiles to approximate the standard deviation for each angular dimension. We then sampled $N = 130$ gaze directions from this distribution and projected each sampled direction into the 2D image space using the same fixed-length projection method. This produced a set of 2D gaze points, from which we computed the empirical standard deviation separately for the x and y axes. These values were used to define **non-isotropic Gaussian blobs** that reflected directional uncertainty in the image plane. A fixed scale factor was applied to the computed variances to ensure the blobs were sufficiently wide, while preserving the relative anisotropy of the gaze distribution. This resulted in input saliency maps that encode gaze uncertainty with variable spread along horizontal and vertical directions.

3.3 Output Map Generation

Given the N fixation points extracted from human observers, we constructed two types of output saliency maps: binary and continuous.

1. **Binary maps**: Each fixation point is represented as a single activated pixel with a value of 1, while all other pixels are set to 0. This sparse representation directly encodes the discrete fixation targets.
2. **Continuous maps**: To generate smooth saliency distributions, we placed a Gaussian blob at each fixation point. Unlike the input maps, here we assumed a **fixed and isotropic** spread across both spatial dimensions. The standard deviation σ is derived from the full width at half maximum (FWHM), set to 1 degree of visual angle, which corresponds to approximately 66 pixels in our setup:

   $$\sigma = \frac{W}{2\sqrt{2\ln 2}}, \quad \text{where } W = 66$$

   yielding the covariance matrix:

   $$\Sigma = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}$$

   This assumption is commonly used in saliency modeling and eye-tracking literature, where 1 degree of visual angle approximates the size of the foveal region and reflects the spatial precision of human fixations. It has been adopted in several benchmark datasets, including the MIT Saliency Benchmark[4], which applies Gaussian smoothing based on this visual angle standard. Each Gaussian blob is weighted by the corresponding **fixation duration**, so that longer fixations contribute more strongly to the saliency map. The final continuous map is obtained by summing all weighted blobs and normalizing the pixel values by the global maximum to preserve peak intensity and ensure consistent scaling.

3.4 Model Architecture and Training Objective

All models are based on a shared convolutional encoder-decoder architecture using a ResNet-18 backbone. The encoder extracts spatial features from the input saliency map, while the decoder progressively upsamples and refines these features to produce output maps. For multi-task models, the decoder branches into two heads to predict both binary and continuous saliency maps simultaneously. Table 1 summarizes the model architecture, detailing the layer types, activation functions, and output shapes. This structure forms the basis for both single-task and multi-task configurations.

Table 1 Model architecture

Layer	**Activation Function**	**Output Shape**
ResNet18 (backbone)	-	[1, 512, 7, 7]
Conv2d(512 → 64, kernel=3, pad=1)	ReLU + BatchNorm	[1, 64, 7, 7]
Upsample(scale=2, bilinear)	—	[1, 64, 14, 14]
Conv2d(64 → 32, kernel=3, pad=1)	ReLU + BatchNorm	[1, 32, 14, 14]
Upsample(scale=4, bilinear)	—	[1, 32, 56, 56]
Conv2d(32 → 16, kernel=3, pad=1)	ReLU + BatchNorm	[1, 16, 56, 56]
First Map Head	—	—
Upsample(scale=4, bilinear)	—	[1, 16, 224, 224]
Conv2d(16 → 1, kernel=1)	Sigmoid	[1, 1, 224, 224]
Second Map Head (if multi-task)	—	—
Upsample(scale=4, bilinear)	—	[1, 16, 224, 224]
Conv2d(16 → 1, kernel=1)	Sigmoid	[1, 1, 224, 224]

To train the models, we used **binary cross-entropy (BCE)** loss for both output types. Following the reasoning in SalGAN [5], saliency values are interpreted as probabilities of attention at each pixel. So we apply an element-wise sigmoid activation, treating each pixel as an independent binary random variable.

In the multi-task setting, we computed two separate BCE losses: one for the binary target and one for the continuous target. Each loss is weighted to account for class imbalance, using a factor derived from the proportion of positive pixels relative to the total number of pixels in the training data. The total loss is defined as:

$$\mathcal{L}_{\text{multi}} = \lambda_{\text{bin}} \cdot \mathcal{L}_{\text{BCE}}^{\text{binary}} + \lambda_{\text{cont}} \cdot \mathcal{L}_{\text{BCE}}^{\text{continuous}}$$

In the multi-task setting, we assigned equal importance to both objectives by setting $\lambda_{\text{bin}} = \lambda_{\text{cont}} = 0.5$. This ensures that each task contributes equally to the overall optimization. In contrast, single-task models use a single BCE loss with full weight ($\lambda = 1$), corresponding to either the binary or continuous target.

4 Experimental set-up

This section outlines the practical details of our training pipeline, including dataset specifications, preprocessing steps, and training parameters for both single-task and multi-task models.

4.1 Dataset

We used the **Coutrot dataset**[7], which consists of 65 videos depicting social scenes with a varying number of participants (average of 4 people per frame). Each video contains approximately 500 frames on average, with a fixed resolution of 720×1280 pixels. The dataset includes annotated facial landmarks for each individual in every frame, enabling precise cropping and gaze estimation.

4.2 Saliency Map Resolution

All saliency maps (both input and output) are initially computed at the original video resolution (720×1280). This ensures that gaze projections and fixation points are accurately represented before any model-specific resizing is applied.

4.3 Data Splitting Strategy

To ensure robust training and evaluation, the dataset was partitioned into training, validation, and test sets at the **video level**. This means that frames from the same video are never shared across different splits, preventing any temporal or contextual leakage.

- **Training set**: 40 videos
- **Validation set**: 11 videos
- **Test set**: 13 videos

A **stratified sampling approach** was used to construct the splits, taking into account two key attributes: the number of faces per frame and the total number of frames per video. These attributes were first analyzed using empirical cumulative distribution functions (ECDFs), from which quantized bins were derived. Stratification was then performed to ensure that each subset (training, validation, test) maintained a balanced representation across these bins. This strategy ensures that the model is exposed to a diverse range of social configurations during training, while preserving statistical consistency across evaluation splits.

4.4 Input Transformations

Prior to feeding frames into the model, each video frame is resized to a fixed resolution of 224×224. After resizing, frames are converted into normalized tensors using standard ImageNet statistics (mean and standard deviation per RGB channel).

4.5 Target Transformations

We defined two distinct preprocessing pipelines for the target saliency maps, depending on whether the task involves continuous or binary output:

- **Continuous maps**: they are resized to match the input resolution using bilinear interpolation. This ensures spatial alignment between input frames and predicted saliency distributions, while preserving smoothness in the output.
- **Binary maps**: due to their sparse nature, are directly generated at the target resolution (224×224) without any resizing. Resizing these maps led to significant degradation, as many closely spaced fixation pixels were lost during interpolation. To avoid this issue, we preserved the original binary structure by applying only tensor conversion, ensuring that fixation points remained intact.

4.6 Training parameters and strategy

We adopted a staged training strategy:

- **Continuous single-task training**: The model was first trained to predict continuous saliency maps using a learning rate of 1×10^{-4} and a batch size of 64. Early stopping was applied based on validation loss, and training converged after 13 epochs. To address class imbalance, a weight of 5 was assigned to positive pixels in the binary cross-entropy (BCE) loss.
- **Binary single-task and multi-task training**: Both models were initialized from the checkpoint of the trained continuous model. The binary single-task model converged after just 2 epochs, while the multi-task model required 8 epochs to reach convergence. For the binary task, a weight of 36 was applied to positive pixels in the BCE loss to compensate for their sparsity.

5 Evaluation

This section presents the evaluation results used to compare the performance of single-task and multi-task models across both saliency prediction tasks: binary and continuous.

5.1 Evaluation Metrics

We employed two complementary metrics to assess model performance:

- **Normalized Scanpath Saliency (NSS)**: Measures how well the predicted saliency map aligns with human fixation points. It is computed as:

$$\text{NSS} = \frac{1}{N} \sum_{i=1}^{N} \hat{S}(x_i)$$

  where $\hat{S}$ is the normalized saliency map (zero mean, unit variance), and x_i are the coordinates of human fixations. NSS is applicable to both binary and continuous predictions.

- **Similarity Metric (SIM)**: Evaluates the structural similarity between the predicted and ground truth saliency maps. It is defined as:

$$\text{SIM} = \sum_i \min(P_i, Q_i)$$

where P and Q are normalized saliency distributions over all pixels. SIM is used exclusively for continuous maps, as it captures the fidelity of spatial structure.

5.2 Quantitative Results

Table 2 shows the per-video scores for each model, task, and metric. To provide a more concise overview, the mean scores across all 13 videos are reported below:

Table 2 Evaluation Scores Across 13 Video Samples

Video ID	NSS Binary	NSS Continuous	NSS Binary Multi	NSS Continuous Multi	SIM Continuous	SIM Continuous Multi
1	3.8156	3.6464	4.1462	3.8036	0.3260	0.4419
2	3.9337	3.4078	3.5608	3.1765	0.3164	0.3773
5	1.4161	1.0061	1.0124	1.1529	0.2725	0.3109
11	1.1073	0.8200	0.9703	1.0679	0.1960	0.2353
22	4.1719	3.8316	3.9783	3.6862	0.3502	0.4249
31	3.3528	3.0901	3.1118	2.9714	0.3291	0.3792
36	1.5512	1.2761	1.2675	1.4549	0.2555	0.2891
37	1.9706	3.2131	2.5494	2.7935	0.2537	0.2870
38	0.8234	1.0183	0.9460	0.9396	0.2731	0.2716
66	1.7525	2.1527	1.4104	1.5666	0.2283	0.2170
68	3.9389	3.5407	3.8775	3.6639	0.4333	0.5279
72	1.7059	1.4336	1.8762	1.9265	0.2317	0.2802
73	4.3781	3.0643	3.8517	3.7134	0.2891	0.4083

- **NSS Binary:** 2.609
- **NSS Continuous:** 2.423
- **NSS Binary Multi:** 2.504
- **NSS Continuous Multi:** 2.455
- **SIM Continuous:** 0.289
- **SIM Continuous Multi:** 0.342

From the results above, it appears that the NSS scores are relatively consistent across models and tasks, indicating comparable performance in terms of saliency prediction. In contrast, the SIM metric shows slightly higher values for the multi-task model, which may suggest a modest improvement in similarity to ground truth maps when multi-task learning is applied.

5.3 Statistical Analysis

To determine whether performance differences between single-task and multi-task models are statistically significant, we conducted paired t-tests for each evaluation metric. The significance level for all tests was set at $\alpha = 0.05$. Prior to hypothesis testing, we assessed the normality assumption using Q–Q plots and the Shapiro–Wilk test. The Shapiro–Wilk test returned p-values above the significance threshold for all metrics, indicating no significant deviation from normality and supporting the use of parametric testing. Three comparisons were performed:

- **Binary map task (NSS metric):** The paired t-test yielded $t = 1.17$, $p = 0.26$, indicating no statistically significant difference between models. The effect size, measured by Cohen's $d = 0.325$, suggests a small effect.
- **Continuous map task (NSS metric):** The result was $t = -0.34$, $p = 0.74$, again showing no significant difference. Cohen's $d = -0.093$ reflects a negligible effect.
- **Continuous map task (SIM metric):** The test produced $t = -4.86$, $p = 0.0004$, revealing a statistically significant difference in favor of the multi-task model. Cohen's $d = -1.347$ indicates a large effect size.

These results suggest that while NSS scores are broadly similar across models, the multi-task approach may offer a meaningful improvement in terms of SIM (for continuous map prediction), indicating better reconstruction of saliency structure.

5.4 Qualitative Analysis

To further interpret model behavior, we constructed a heatmap of normalized scores across all videos, models, and evaluation metrics. This visualization helps identify easy and hard samples (i.e., videos where models consistently perform well or poorly). The heatmap is shown in Figure 2. Based on the heatmap of normalized scores, certain video samples exhibit consistent performance trends across models and metrics. In particular, video 68 and video 73 appear to be relatively easier, with consistently higher similarity scores across most metrics. Conversely, video 11 and video 38 show lower scores across the board, suggesting they are more challenging for all models. To explore whether spatial variability in binary target maps correlates with task difficulty, we computed a global variability score for selected video samples. Specifically, we sampled 100 frames each from a subset of videos identified as either relatively easy (IDs: 68, 1, 2, 73) or hard (IDs: 36, 11, 5, 38), based on their normalized performance scores. For each binary target map, we calculated the spatial variability as the Euclidean norm of the standard deviations along the x and y axes:

$$\sigma_{\text{spatial}} = \sqrt{\sigma_x^2 + \sigma_y^2}$$

This metric captures the overall dispersion of fixation points across the image. While this analysis does not cover the entire test set, it provides a focused comparison between extremes. The median variability score was notably higher for hard videos (51.88) compared to easy ones (41.19), suggesting that increased spatial dispersion in fixation patterns may be associated with reduced model performance. These findings support the hypothesis that harder samples exhibit less concentrated attention, making saliency prediction more challenging. To further illustrate model behavior, we present two representative examples drawn from the extremes of the performance spectrum identified in the normalized score heatmap (Figure 2). Figure 3 shows prediction outputs for an easier video (73), where both continuous and binary tasks yield

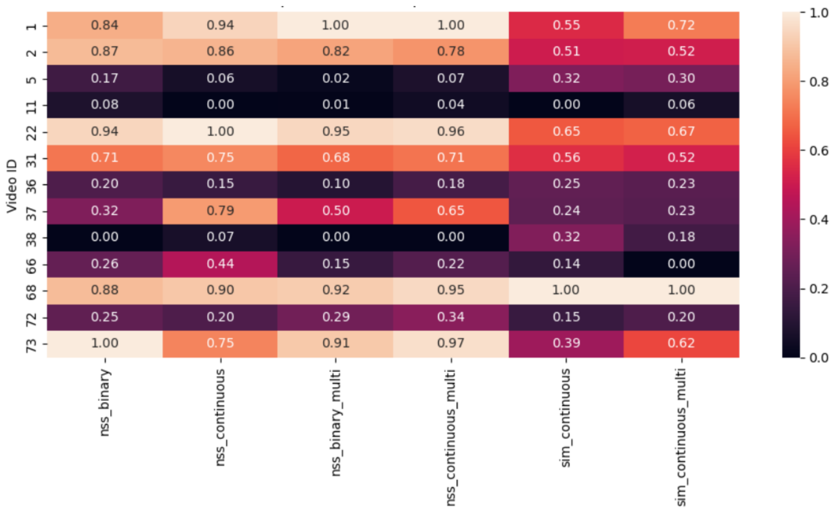

Fig. 2 Heatmap of normalized scores across videos, models, and metrics. Darker regions indicate lower performance, while lighter regions highlight stronger results.

predictions that closely resemble their respective ground truth targets. High-intensity regions are well aligned, indicating that the models effectively captured salient patterns. In contrast, Figure 4 displays predictions for a harder video sample (11). Here, the continuous prediction diverges noticeably from the target in both spatial distribution and intensity, while the binary prediction fails to localize key regions, resulting in sparse or misaligned activations. These differences underscore the variability in model performance across video samples and highlight the challenges posed by certain content.

6 Conclusion

In this work, we explored the effectiveness of single-task and multi-task learning strategies for saliency prediction across binary and continuous tasks. Quantitative evaluation using NSS and SIM metrics revealed broadly comparable performance across models, with NSS scores showing minimal variation. However, the SIM metric, used to assess structural similarity in continuous predictions, highlighted a statistically significant advantage for the multi-task model, supported by a large effect size in paired t-tests. Overall, our results suggest that while single-task models perform adequately, multi-task learning offers tangible benefits in capturing the structural nuances of saliency.

References

[1] Jeffrey T. Klein1, Stephen V. Shepherd2 and Michael L. Platt1: Social Attention and the Brain,2009

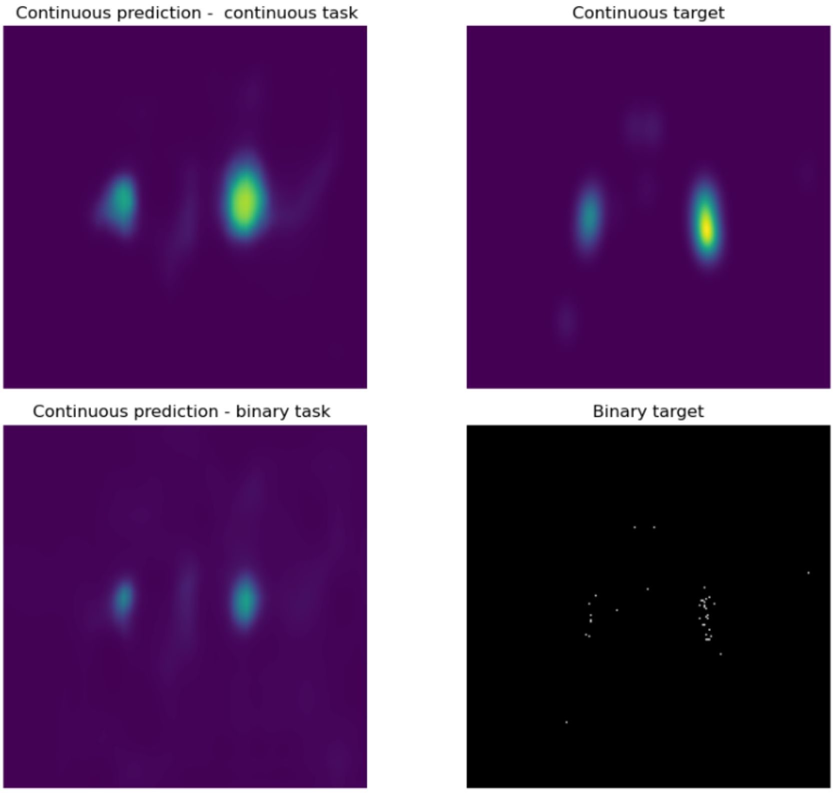

Fig. 3 Prediction outputs for an easier video sample. Top: continuous prediction and target; bottom: binary prediction and target.

[2] Zoya Bylinskii , Tilke Judd, Aude Oliva, Antonio Torralba, and Fredo Durand: What D o Different Evaluation Metrics Tell Us About Saliency Models? 2019

[3] Yasser Abdelaziz Dahou Djilali1,2 Kevin McGuinness 1 Noel O'Connor1 1Dublin City University, Ireland: Learning Saliency From Fixation

[4] MIT CSAIL: MIT Saliency Benchmark. Available at http://saliency.mit.edu/datasets.html, 2015.

[5] Junting Pana, Cristian Canton-Ferrerb, Kevin McGuinnessc, Noel E. O'Connorc, Jordi Torresd, Elisa Sayrola, Xavier Giro-i-Nieto: SalGAN: visual saliency prediction with adversarial networks, 2018

[6] Petr Kellnhofer 1, Adri'a Recasens1, Simon Stent2, Wojciech Matusik1: Gaze360: Physically Unconstrained Gaze Estimation in the Wild

[7] Erwan David, Jesus Gutierrez, Antoine Coutrot: Erwan David, Jesus Gutierrez, Antoine Coutrot, 2019

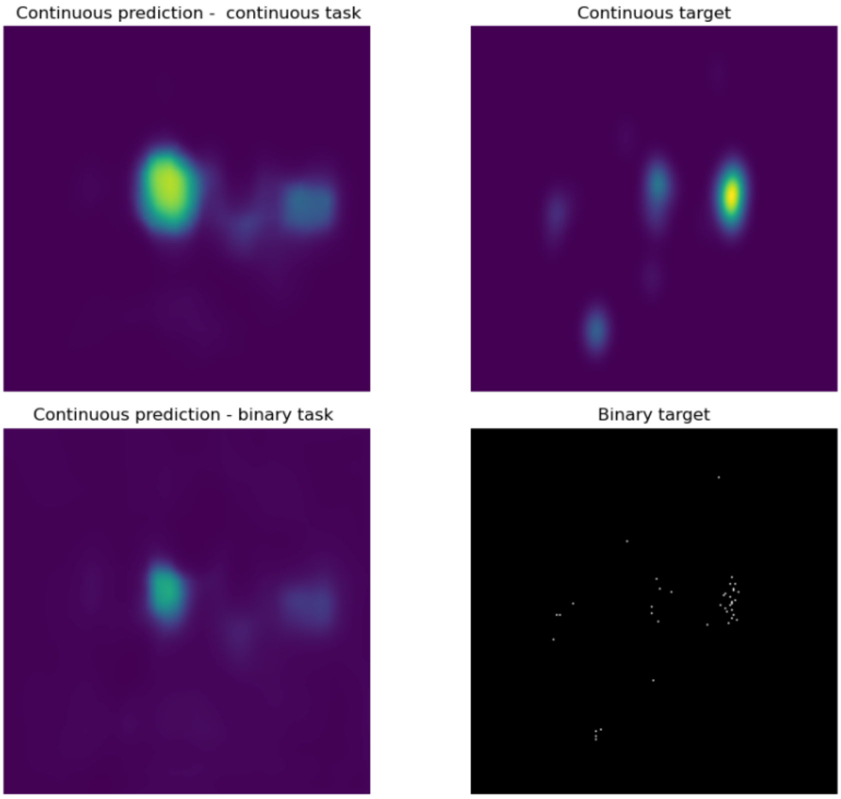

Fig. 4 Prediction outputs for a harder video sample. Top: continuous prediction and target; bottom: binary prediction and target.