

# Noise-Aware Video Saliency Prediction - Supplementary Material

Ekta Prashnani<sup>1,2</sup>

ekta@ece.ucsb.edu

Orazio Gallo<sup>2</sup>

ogallo@nvidia.com

JooHwan Kim<sup>2</sup>

sckim@nvidia.com

Josef Spjut<sup>2</sup>

jspjut@nvidia.com

Pradeep Sen<sup>1</sup>

psen@ece.ucsb.edu

Iuri Frosio<sup>2</sup>

ifrosio@nvidia.com

<sup>1</sup> University of California,

Santa Barbara,

California, USA

<sup>2</sup> NVIDIA Research,

Santa Clara,

California, USA

## 1 Derivation of NAT cost function (mentioned in Sec. 3)

We interpret  $d(x_i, \tilde{x}_i)$  as a random variable with Gaussian distribution,  $d(x_i, \tilde{x}_i) \sim G(\mu_i, \sigma_i^2)$ , where  $\mu_i = E[d(x_i, \tilde{x}_i)]$  indicates its mean, whereas  $\sigma_i^2 = \text{Var}[d(x_i, \tilde{x}_i)]$  is its variance. When the predicted saliency map  $\hat{x}_i$  is optimal, *i.e.* when  $\hat{x}_i = x_i$ ,  $d(\hat{x}_i, \tilde{x}_i)$  has the same statistical distribution of  $d(x_i, \tilde{x}_i)$ . Therefore, for a perfect saliency predictor, we can write  $d(\hat{x}_i, \tilde{x}_i) \sim G(\mu_i, \sigma_i^2)$ . Note that, for our proposed noise-aware training (NAT),  $\mu_i$  and  $\sigma_i$  are assumed to be known, and therefore,  $\hat{x}_i$  is the only unknown. The likelihood of  $d(\hat{x}_i, \tilde{x}_i)$  is given by:

$$p[d(\hat{x}_i, \tilde{x}_i)] = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{[d(\hat{x}_i, \tilde{x}_i) - \mu_i]^2}{2\sigma_i^2}}. \quad (1)$$

Given our interpretation of  $d(\hat{x}_i, \tilde{x}_i)$ , for a dataset containing  $N + 1$  saliency maps, the negative log likelihood is:

$$\begin{aligned} J(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N) &= -\ln \prod_i \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{[d(\hat{x}_i, \tilde{x}_i) - \mu_i]^2}{2\sigma_i^2}} = \\ &= \sum_i -\ln \left\{ \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{[d(\hat{x}_i, \tilde{x}_i) - \mu_i]^2}{2\sigma_i^2}} \right\} = \\ &= \sum_i \left\{ \ln(\sqrt{2\pi}\sigma_i) + \frac{[d(\hat{x}_i, \tilde{x}_i) - \mu_i]^2}{2\sigma_i^2} \right\}. \end{aligned} \quad (2)$$

We want to train the saliency models to predict all the  $\{\hat{x}_i\}_{i=0\dots N}$  that maximize the likelihood. Therefore, the optimization problem becomes:

$$\begin{aligned} (\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N) &= \underset{(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N)}{\operatorname{argmin}} J(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N) = \\ &\underset{(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N)}{\operatorname{argmin}} \sum_i \left\{ \ln(\sqrt{2\pi}\sigma_i) + \frac{[d(\hat{x}_i, \tilde{x}_i) - \mu_i]^2}{2\sigma_i^2} \right\}. \end{aligned} \quad (3)$$

Upon simplification (removing the terms that do not depend on  $(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N)$ , that are the only unknowns), we obtain:

$$(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N) = \underset{(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N)}{\operatorname{argmin}} \sum_i \left\{ \frac{[d(\hat{x}_i, \tilde{x}_i) - \mu_i]^2}{\sigma_i^2} \right\}. \quad (4)$$

This leads to the formulation of the NAT cost function:

$$J_{\text{NAT}}^{\text{ideal}} = \sum_i \left\{ \frac{[d(\hat{x}_i, \tilde{x}_i) - \mu_i]^2}{\sigma_i^2} \right\} = \sum_i \left\{ \frac{[d(\hat{x}_i, \tilde{x}_i) - \mathbb{E}[d(x_i, \tilde{x}_i)]]^2}{\text{Var}[d(x_i, \tilde{x}_i)]} \right\}. \quad (5)$$

## 2 A toy example to motivate NAT (mentioned in Sec. 3)

Assume that a method predicts the (unobservable) distribution  $x_i$  exactly, that is  $\hat{x}_i = x_i$ . Because of measurement noise and incomplete sampling in  $\tilde{x}_i$  (which is the saliency map estimated from insufficient gaze data, *i.e.* the one typically used for training),  $d(x_i, \tilde{x}_i) \neq 0$ , even though the prediction is perfect. In this scenario, it would suboptimal to train a saliency predictor to minimize  $d(x_i, \tilde{x}_i)$ .

Let us consider a 1D toy example: Figs. 1(a,h) show two 1D ground-truth “saliency maps” (or pdfs)  $x_i$ , one unimodal, and one bimodal. We simulate the “1D gaze-data acquisition” by sampling 3 (red circles) or 30 (blue) spatial locations (or “gaze fixations”) from  $x_i$ . Following the *de facto* standard to generate saliency maps from single gaze locations, we blur each fixation (Fig. 1(b)), and accumulate the resulting curves (Fig. 1(c)). This results in approximations,  $\tilde{x}_i$ , of the ground-truth saliency maps. The inaccurate positions of the modes in these estimated saliency maps mimics the measurement noise, while the finite number of 1D gaze fixations used to estimate these maps simulates incomplete sampling.

When few fixations are available,  $\tilde{x}_i$  may be shifted with respect to  $x_i$  (Fig. 1(c)), and the number of its modes may not match  $x_i$  (Fig. 1(j)). Furthermore, when  $x_i$  is multimodal, the mass of each mode in  $\tilde{x}_i$  may be imprecisely estimated compared to  $x_i$  (Fig. 1(j)). The standard deviation of 1000 random realizations of  $\tilde{x}_i$  ( $\text{Std}[\tilde{x}_i]$ ), which measures the uncertainty in  $\tilde{x}_i$  (and therefore the quality of estimation of  $x_i$  using  $\tilde{x}_i$ ), decreases when a large number of fixations are used to reconstruct  $\tilde{x}_i$  and remains high for a smaller number of fixations. This is shown as the light-blue / light-red shaded regions in Figs. 1(d, g, k, n), while the solid plot red / blue curve shows  $\mathbb{E}[\tilde{x}_i]$ . Furthermore, the level of uncertainty is proportional the complexity of the ground-truth saliency map: *e.g.*, given 3 fixations to reconstruct  $\tilde{x}_i$ , the uncertainty is lower when the underlying ground-truth  $x_i$  map is unimodal (Fig. 1(d)), and higher when  $x_i$  is bimodal Fig. 1(k). We note that in Figs. 1(d, g, k, n),  $\mathbb{E}[\tilde{x}_i]$  still differs from  $x_i$  because of the blurring operation used in the reconstruction of  $\tilde{x}_i$  from sampled 1D locations from  $x_i$ . When the reconstruction process for  $\tilde{x}_i$  is perfect (a topic of research beyond the scope of this work), such reconstruction errors would be eliminated. For our experiments, we adopt this standard reconstruction process.

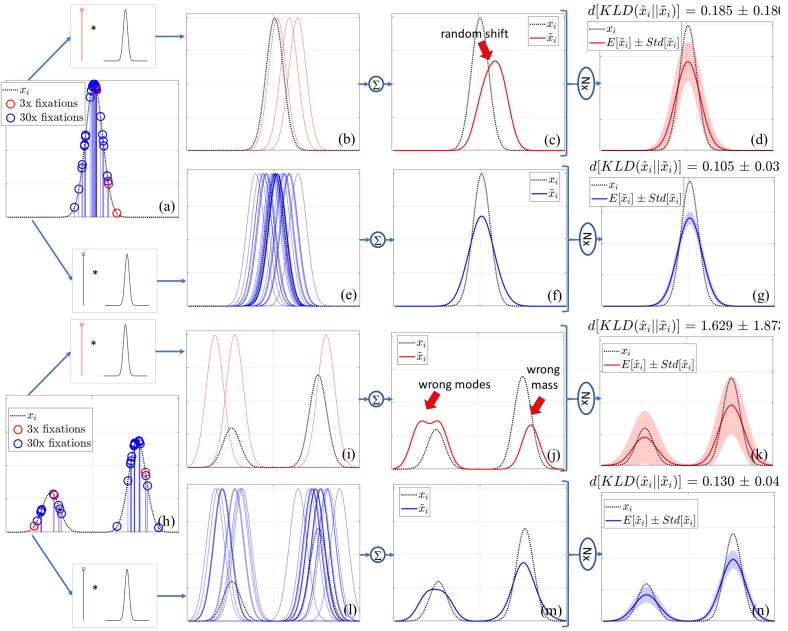


Figure 1: A toy example to motivate NAT. Plots in (a) and (h) show the unimodal and multimodal 1D pdfs  $x_i$  in dashed black lines – these are analogous to the true underlying 2D saliency maps for video frames / images. The measured (or training) saliency maps are reconstructed by first sampling “fixations” (red / blue circles in (a, h)) from  $x_i$ , then blurring (b, e, i, l), summing, and normalizing to obtain the resulting reconstructed saliency maps  $\tilde{x}_i$  (d, g, j, m). When a limited number of observers is available (e.g., 3, in the red plots in (c, j)), the resulting reconstructed  $\tilde{x}_i$  may differ in shape from  $x_i$ , e.g., due to random shifts, reconstruction errors, etc. Plots d, g, k, n show the expected value and standard deviation for multiple realizations of  $\tilde{x}_i$ , with respect to  $x_i$ . The deviation of  $\tilde{x}_i$  from  $x_i$  results in the statistics  $\mathbb{E}[\text{KLD}(x_i, \tilde{x}_i)]$  and  $\text{Var}[\text{KLD}(x_i, \tilde{x}_i)]$  to be non-zero (as shown in the titles of plots d, g, k, n). Furthermore, as is evident from these plots, these statistics are larger when few observers are available and when  $x_i$  has a complex shape (e.g., multimodal), which makes  $x_i$  more susceptible to inaccurate approximation using  $\tilde{x}_i$ . The plots considered here for  $x_i$  are: a Gaussian centered at  $\mu = 50$ ,  $\sigma = 5$ ; and a mixture with two components at  $\mu = [25, 75]$ , probabilities  $P = [0.3, 0.7]$ , and  $\sigma = 5$ .

The uncertainty in  $\tilde{x}_i$  due to measurement noise and incomplete sampling results in uncertainty in accurately estimating  $d(x_i, \tilde{x}_i)$ . We now want to estimate the distribution  $p[d(x_i, \tilde{x}_i)]$ , where we model  $d(x_i, \tilde{x}_i)$  as a Gaussian random variable. We compute  $\text{KLD}(x_i, \tilde{x}_i)$  for 1,000 random realizations of  $\tilde{x}_i$  and estimate  $\mathbb{E}[\text{KLD}(x_i, \tilde{x}_i)]$ ,  $\text{Std}[\text{KLD}(x_i, \tilde{x}_i)]$ . These are reported in the titles of Figs. 1(d, g, k, n). We use KLD as discrepancy function because of its wide adoption for saliency estimation, but the results presented here hold for other metrics as well. We observe that:

- $\mathbb{E}[\text{KLD}(x_i, \tilde{x}_i)] > 0$ , i.e.  $\text{KLD}(x_i, \tilde{x}_i)$  is biased. The source of the bias is twofold. First,  $\text{KLD}(x_i, \tilde{x}_i) > 0$  because  $\mathbb{E}[\tilde{x}_i]$  is a smoothed version of  $x_i$  (bias due to the choice of the method used to reconstruct  $\tilde{x}_i$ ), independently from the number of observers.

Second,  $\tilde{x}_i$  is noisy ( $\text{Std}[\tilde{x}_i] > 0$ ), which, especially for a limited number of observers, contributes with an additional bias to  $\text{KLD}(x_i, \tilde{x}_i)$ .

- $\text{Std}[\text{KLD}(x_i, \tilde{x}_i)] > 0$ , and it tends to be smaller for a larger number of observers.
- For a given number of observers,  $\mathbb{E}[\text{KLD}(x_i, \tilde{x}_i)]$  and  $\text{Std}[\text{KLD}(x_i, \tilde{x}_i)]$  are larger for multimodal maps.

We conclude that, when  $\tilde{x}_i$  is affected by measurement noise and incomplete sampling, the expected value and variance of the discrepancy  $d(x_i, \tilde{x}_i)$  are not zero, depend on the number of observers, and are different for each frame. These properties, which also hold for 2D saliency maps recorded from real human observers, form the basis for the development and interpretation of NAT.

### 3 Gaze data analysis for ForGED (mentioned in Sec. 4)

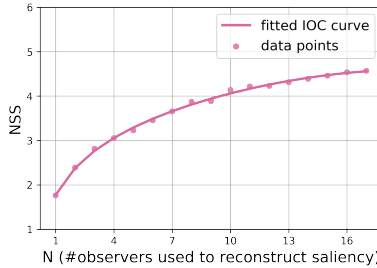


Figure 2: Inter-observer consistency (IOC) curve computed on test-set frames of ForGED containing at least 19 observers. Each data point is an average of the IOC value for the given value of  $N$ , with the average computed over multiple realizations across the frames. The fitted curve is shown with a solid line and indicates the diminishing amount of new information that gaze data from additional observers imparts, when  $N$  is sufficiently high.

**Observer consistency and ForGED dataset split.** As discussed in Sec. 1 of the main paper, IOC curve measures how well a saliency map reconstructed from gaze data of  $N$  observers explains the gaze of a new observer as a function of  $N$  [5, 8, 14]. A converged IOC curve indicates that additional observers do not add significant new information to the reconstructed saliency map [5, 14]. A typical test of whether a dataset captures sufficient observers is to evaluate the level of convergence of the IOC curves *on average across all frames* at maximum value for  $N$  (sometimes by using curve-fitting and extrapolation [14]). To obtain the average IOC for ForGED, we sample 1 out of every 5 frames from ForGED test videos containing at least 19 observers – for a total of 1500 frames. For each frame, we compute the per-frame IOC curve with 20 random realizations for the subset of observers that constitute the  $N$  observers and the subset that constitutes the new observer whose gaze data is to be explained by the  $N$ -observer saliency map. All realizations of the IOC curves across all sampled frames are averaged to obtain the IOC curve shown in Fig. 2. As can be seen from Fig. 2, the gradient magnitude of the IOC curve is small at  $N = 17$  (0.04). This further diminishes upon extrapolation to  $N = 21$  observers to 0.02. Our test set therefore



contains gaze data from up to 21 observers per video (median 17). As noted in the main paper (Limitations and Future Work in Sec. 6) - while on average the IOC curves across all evaluated datasets (LEDOV, DIEM, ForGED) show very small gradient at sufficiently high number of observers, the level of convergence for each frame may be different (content-dependent) and motivates the need for NAT. This also presents an interesting direction of future research to design noise-robust evaluation schemes. Note that, while the ForGED test dataset contains gaze data from a large number of observers (that ensures small gradients in the IOC curves at maximum available  $N$ ), the ForGED training dataset consists of a larger number of videos but with gaze data from only 5 – 15 observers (the majority of the videos contain 5 observers). This setting simulates the scenario where training data with limited number of observers is available (the setting most suitable for NAT) – while the testing is always performed on more accurate saliency maps. The training-validation-test split for ForGED videos is 379 videos for training, 26 for validation, and 75 for testing. As already discussed in the main paper (and also shown in Sec. 5), for the different experiments enlisted in tables, the training dataset size is varied in terms of number of available training videos,  $V$ , and number of observers,  $N$ , used to reconstruct the saliency maps  $\tilde{x}_i$  per video – to demonstrate the performance gain of NAT for varying amount of training data.

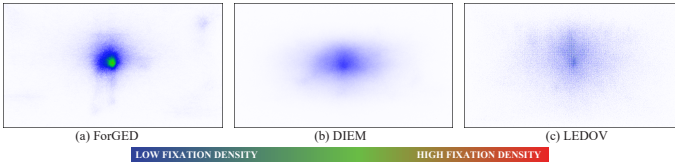


Figure 3: Accumulated fixation density across gaze data from all observers across all frames in (a) ForGED (b) DIEM and (c) LEDOV.

**Observer gaze behavior in ForGED.** Given that the main character is placed at the center of the screen in Fortnite game, we observe an affinity towards center in the gaze behavior. Events such as combat, focused motion towards a location such as the horizon, attempts to pick up resources such as ammunitions lead to observer gaze behavior that follows the narrative of the game-play (*e.g.*, viewers observe the opponent when the main character is in combat, viewers look towards the horizon when the main character is moving towards it). On the other hand, when a scene becomes relatively uneventful, such as when the main central character has been running towards the horizon for a few seconds, the observers’ gaze tends to become more exploratory – scanning the surroundings, or simply browsing the evolving scenery. Examples of all such scenes can be found in the supplementary video, Fig. 3 of the main paper, and Fig. 4 in this Supplementary document. Lastly, we accumulate all of the gaze locations captured on ForGED into a fixation density map (Fig. 3a) to assess the common viewing tendencies of observers. We also compare these for LEDOV and DIEM. As discussed in Sec. 4 of the main paper, due to the main character near the center of the frame, the aiming reticle at the center of the frame, and a guiding mini-map on the top right, observers look at these regions frequently. As compared to LEDOV and DIEM, such a behavior is uniquely representative of the observer behavior in third person shooting games such as Fortnite. In case of LEDOV and DIEM, we also observe a bias towards the center – but it tends to be more widespread as shown in Fig. 3.

## 4 Analysis of approximation in Eq. 6 (mentioned in Sec. 3)

To analyze the accuracy of Eq. 6 in the main paper, we select a subset of the video frames from the DIEM dataset that contains gaze data from more than 200 observers. Given the very large number of gaze fixations for these frames, we anticipate that the estimated human-saliency map  $\tilde{x}_i$  is very close to ground-truth saliency  $x_i$  [14] for every such frame  $i$  (as also confirmed by converged IOC curves for these frames). We therefore analyze the accuracy of Eq. 6 under the assumption that the  $> 200$ -observer gaze maps of these frames represent  $x_i$ . From these 200-observer gaze maps ( $x_i$ ), we sample a certain number (denoted as  $M$ ) of gaze fixation locations followed by blurring to compute  $\tilde{x}_i$ . Therefore,  $\tilde{x}_i = SR(x_i; M)$ . Then, we compute  $\tilde{\tilde{x}}$  by sampling  $M$  spatial locations as per the pdf  $\tilde{x}$  followed by blurring. That is,  $\tilde{\tilde{x}} = SR(\tilde{x}; M)$ .

Using multiple realizations of  $\tilde{x}$  and  $\tilde{\tilde{x}}$ , we estimate  $\mathbb{E}[d(x, \tilde{x})]$ ,  $\mathbb{E}[d(\tilde{x}, \tilde{\tilde{x}})]$ ,  $\text{Var}[d(x, \tilde{x})]$ ,  $\text{Var}[d(\tilde{x}, \tilde{\tilde{x}})]$ . We find that the mean absolute percentage error (MAPE) in the approximation of  $\mathbb{E}[d(x, \tilde{x})]$  (Eq. 6 in main paper) goes from 21% for  $N = 5$ , to 13% for  $N = 15$ , and down to 10% for  $N = 30$ . Similarly, MAPE in the approximation of  $\text{Var}[d(x, \tilde{x})]$  (Eq. 6 in main paper) goes from 13% for  $N = 5$ , to 6% for  $N = 15$ , and down to 5% for  $N = 30$ . Note that a large under/over-estimation of  $\mathbb{E}[d(x, \tilde{x})]$  and  $\text{Var}[d(x, \tilde{x})]$  in Eq. 6 (main paper) may lead to overfitting to noisy data or sub-optimal convergence respectively using Eq. 7 (main paper) for training. This would result in poor performance of NAT compared to traditional training – which, as shown by the results, is not the case.

## 5 Additional Results (mentioned in Sec. 5)

We now report the additional experiments performed to compare NAT (Eq. 6 in main paper) to traditional training (abbreviated as TT in this section; Eq. 2 in main paper). Furthermore, we show typical gaze maps obtained through TT and NAT compared to the ground truth for TASED on the ForGED dataset in Fig. 4.

### 5.1 Dataset type and size

In this section, we continue reporting the results from Sec. 5 of the main paper, where we compared NAT vs. TT for different dataset types and sizes. Table 1 compares the performance of TT to NAT on an additional dataset, the DIEM dataset [14], for the TASED architecture [14], and using KLD as discrepancy for training. As done throughout Sec. 5 of the main paper, the evaluation is performed on videos with gaze data from *all* of the available observers (in contrast to training, for which a subset of observers are used, see Table 1 in main paper). In case of DIEM dataset, given that only 84 videos are available, we use 30 or 60 videos for training and report the results on the remaining 24 videos, which are also used as validation set. The number of observers for these videos ranges from 51 to 219, which makes DIEM a very low-noise evaluation set [14]. Results on DIEM are consistent with those reported in the main paper, with NAT providing better metrics in evaluation when compared to TT when less training data (e.g., 30 videos) is available.

train videos $V$	train obs. $N$	loss	KLD↓	CC↑	SIM↑	NSS↑	AUC-J↑
30	5	TT	0.641	0.698	0.591	<b>3.517</b>	0.922
		NAT	<b>0.599</b>	<b>0.708</b>	<b>0.592</b>	3.513	<b>0.934</b>
	15	TT	0.597	0.710	0.602	3.582	0.930
		NAT	<b>0.583</b>	<b>0.718</b>	<b>0.607</b>	<b>3.627</b>	<b>0.932</b>
	31	TT	0.576	0.724	0.614	3.663	0.925
		NAT	<b>0.559</b>	<b>0.731</b>	<b>0.618</b>	<b>3.694</b>	<b>0.928</b>
60	5	TT	0.528	0.735	<b>0.619</b>	<b>3.709</b>	0.933
		NAT	<b>0.518</b>	<b>0.737</b>	0.616	3.639	<b>0.940</b>
	15	TT	<b>0.485</b>	<b>0.757</b>	<b>0.639</b>	<b>3.795</b>	0.933
		NAT	0.493	0.754	0.635	3.792	<b>0.936</b>
	31	TT	0.476	0.759	0.641	3.821	<b>0.938</b>
		NAT	<b>0.467</b>	<b>0.766</b>	<b>0.654</b>	<b>3.864</b>	0.935

Table 1: Saliency metrics on DIEM, for TASED Net, training with KLD as discrepancy, and various number of training videos and observers. The best metrics between TT (Eq. 2 in main paper) and NAT are in bold.

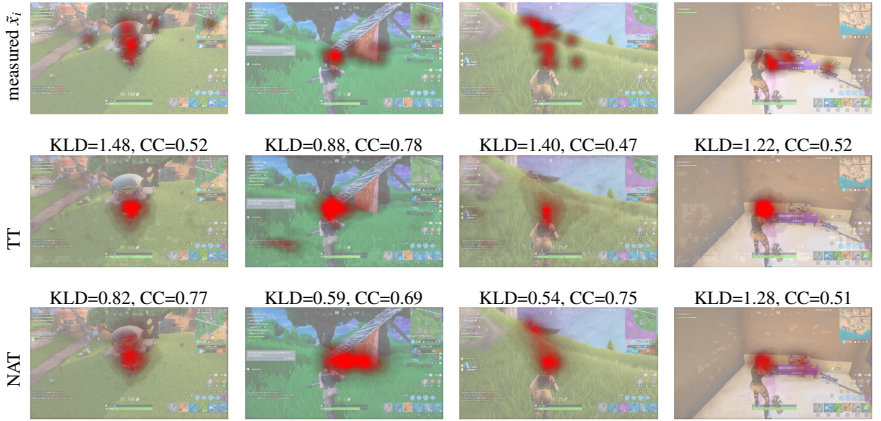


Figure 4: Typical gaze maps obtained through TT (second row – Eq. 2 in main paper) and NAT (third row) compared to the ground truth (first row) for TASED on the ForGED dataset, training with KLD loss, 30 training videos and 5 observers per frame (see Table 4a in main paper). Each panel reports in the title the corresponding KLD and CC values. The last column shows a failure case where the metrics KLD and CC indicate that NAT is worse than TT, although a visual inspection might indicate otherwise. Furthermore, the saliency maps predicted with TT indicate more centralized unimodal predictions – while NAT accurately predicts decentralized, multi-modal saliency maps even when trained with less data. The visualization of saliency map overlays follows the scheme in Fig. 3 of main paper. *ForGED images have been published with permission of Epic Games.*

## 5.2 Discrepancy functions

Table 2 shows NAT vs. TT using  $d = -\text{NSS}$  on ForGED dataset. In Table 2, we notice that NAT overcomes TT in terms of NSS only for 2 or 5 observers, and 30 training videos. Recall that, by design, NSS optimizes the predicted saliency map only at the measured fixation locations. Consequently, when few fixations per frame are available for training, a high NSS score may not generalize well to other evaluation metrics that evaluate different aspects of the quality of a predicted saliency map. This can be alleviated by additional regularization (such as using additional metrics as we do with  $d = \text{KLD} - 0.1\text{CC} - 0.1\text{NSS}$  in Table 4b of the main paper and observe that high NSS scores generalize to good performance in terms of other metrics). In other words, for few-observer training, optimizing for NSS alone may not constrain the predicted saliency map sufficiently — which shows up as poor generalization

to other metrics. This is what we observe in Table 2, where the regularizing effect of NAT leads to worse NSS values compared to TT; but, *all* of the other evaluation metrics indicate NAT to be better.

train videos $V$	train obs. $N$	loss	KLD $_{\downarrow}$	CC $\uparrow$	SIM $\uparrow$	NSS $\uparrow$	AUC-J $\uparrow$
30	2	TT	2.005	0.362	0.302	2.677	0.788
		NAT	<b>1.408</b>	<b>0.528</b>	<b>0.371</b>	<b>3.163</b>	<b>0.887</b>
	5	TT	1.642	0.489	0.28	3.212	0.898
		NAT	<b>1.254</b>	<b>0.566</b>	<b>0.417</b>	<b>3.39</b>	<b>0.906</b>
	15	TT	1.518	0.506	0.403	<b>3.672</b>	0.826
		NAT	<b>1.155</b>	<b>0.608</b>	<b>0.435</b>	3.552	<b>0.91</b>
100	2	TT	1.328	0.563	0.383	<b>3.783</b>	0.899
		NAT	<b>1.206</b>	<b>0.584</b>	<b>0.426</b>	3.44	<b>0.906</b>
	5	TT	1.312	0.578	0.452	<b>3.983</b>	0.835
		NAT	<b>1.165</b>	<b>0.61</b>	<b>0.475</b>	3.747	<b>0.879</b>
	15	TT	1.163	0.614	0.475	<b>4.161</b>	0.857
		NAT	<b>1.028</b>	<b>0.642</b>	<b>0.495</b>	3.858	<b>0.901</b>
379	2	TT	1.093	0.633	0.491	<b>4.381</b>	0.875
		NAT	<b>1.006</b>	<b>0.658</b>	<b>0.512</b>	3.928	<b>0.892</b>
	5	TT	1.093	0.633	0.491	<b>4.381</b>	0.875

Table 2: NAT vs. TT on ForGED for TASED,  $d = -\text{NSS}$  (a fixation-based discrepancy), various number of training videos and observers. Best metrics for each pair of experiments in bold.

To further verify that NAT generalizes to different discrepancy functions, we train and test TASED on LEDOV [9] with the fixation-based discrepancy function,  $d = -\text{NSS}$ , and the combination of fixation and density-based discrepancy functions,  $d = \text{KLD} - 0.1\text{CC} - 0.1\text{NSS}$  (which is a popular discrepancy function used in video-saliency research [9, 14]). The test set for LEDOV is used for all reported evaluations on LEDOV dataset, which contains gaze data from 32 observers per video.

Table 3 shows NAT vs. TT (Eq. 2 in main paper) using  $d = -\text{NSS}$ . For this specific experiment, with TT we observe that adopting RMSprop as the optimizer (as done for all experiments in the paper) shows very fast convergence to very high NSS values. While this property of fast and optimal convergence of discrepancy function has proven useful for all experiments in the paper (see Sec. 6 for details), for this specific experiment the solution provided by RMSprop optimization shows poor generalization to all other saliency metrics. This behavior is alleviated to some extent by switching RMSProp with Stochastic Gradient Descent (SGD) for TT – but at the cost of poor convergence in terms of NSS. To show this, in Table 3, we report two sets of experiments for TT for each size of training dataset (one with SGD and another with RMSprop). With NAT, however, we observe a consistent optimal convergence due to the regularizing effect of the NAT formulation that prevents overfitting to dataset noise.

We further observe that using additional terms with NSS in the discrepancy function, such as with  $d = \text{KLD} - 0.1\text{CC} - 0.1\text{NSS}$  overcomes some of the issues of training with NSS alone. Table 4, 5 show the comparison of TT vs. NAT for this combined discrepancy function. A high NSS performance in this case is well-correlated with good performance in terms of other metrics. Furthermore we note that the performance of NAT is superior to TT when less gaze data is available, with the gap between the two approaches closing in with more gaze data. Given our analyses of all of the experiments with various discrepancy functions and dataset types, our conclusion is that the performance of models trained with density-based discrepancy functions (e.g., KLD) is better for TT as well as NAT, with NAT showing consistent superior performance compared to TT.

train videos $V$	train obs. $N$	loss	KLD↓	CC↑	SIM↑	NSS↑	AUC-J↑
100	5	TT, SGD	2.352	<i>0.244</i>	<b>0.267</b>	2.272	<i>0.761</i>
		TT, RMSprop	4.139	0.192	0.056	<b>9.92</b>	0.178
		NAT	<b>1.746</b>	<b>0.428</b>	<i>0.230</i>	2.358	<b>0.916</b>
	30	TT, SGD	<i>2.302</i>	<i>0.258</i>	<b>0.275</b>	<i>2.661</i>	<i>0.775</i>
		TT, RMSprop	3.593	0.247	0.111	<b>13.628</b>	0.423
		NAT	<b>1.903</b>	<b>0.398</b>	<i>0.198</i>	2.370	<b>0.919</b>
461	5	TT, SGD	2.777	<i>0.317</i>	<i>0.232</i>	<i>4.464</i>	<i>0.612</i>
		TT, RMSprop	4.00	0.241	0.062	<b>14.617</b>	0.206
		NAT	<b>1.305</b>	<b>0.575</b>	<b>0.354</b>	3.29	<b>0.929</b>
	30	TT, SGD	2.252	<i>0.470</i>	<b>0.355</b>	2.463	<i>0.593</i>
		TT, RMSprop	3.526	0.292	0.127	<b>14.048</b>	0.381
		NAT	<b>1.402</b>	<b>0.571</b>	<i>0.310</i>	2.933	<b>0.927</b>

Table 3: Comparison of TT (Eq. 2 in main paper) vs. NAT on LEDOV testing set, for TASED Net, trained with  $-0.1\text{NSS}$  as discrepancy, and various number of training videos and observers. The best metric between each set of 3 experiments for a given dataset size (videos and observers) is in bold and the second-best is italicized. Given the strong overfitting behavior of NSS with TT using RMSprop for this particular set of experiments, we report TT optimized with SGD as well.

train videos $V$	train obs. $N$	loss	KLD↓	CC↑	SIM↑	NSS↑	AUC-J↑
30	30	TT	1.652	0.446	0.261	2.269	0.871
		NAT	<b>1.243</b>	<b>0.494</b>	<b>0.394</b>	<b>2.491</b>	<b>0.900</b>
100	5	TT	1.368	0.496	0.395	2.430	0.863
		NAT	<b>1.149</b>	<b>0.540</b>	<b>0.423</b>	<b>2.782</b>	<b>0.905</b>
	30	TT	1.261	0.534	0.368	2.658	0.903
		NAT	<b>1.034</b>	<b>0.574</b>	<b>0.432</b>	<b>3.250</b>	<b>0.928</b>
	5	TT	1.159	0.577	0.485	<b>3.912</b>	0.864
		NAT	<b>0.852</b>	<b>0.626</b>	<b>0.513</b>	3.451	<b>0.931</b>
461	30	TT	0.913	0.626	0.513	<b>5.743</b>	0.910
		NAT	<b>0.755</b>	<b>0.688</b>	<b>0.554</b>	3.559	<b>0.930</b>

Table 4: Saliency quality metrics on LEDOV testing set, for TASED Net, training with  $\text{KLD}-0.1\text{CC}-0.1\text{NSS}$  as discrepancy, and various number of training videos and observers. The best metrics between TT (Eq. 2 in main paper) and NAT are in bold.

train videos $V$	train obs. $N$	loss	KLD↓	CC↑	SIM↑	NSS↑	AUC-J↑
30	15	TT	0.687	0.696	0.590	3.618	0.900
		NAT	<b>0.588</b>	<b>0.718</b>	<b>0.601</b>	<b>3.629</b>	<b>0.932</b>
	31	TT	<b>0.555</b>	0.727	0.609	3.605	<b>0.935</b>
		NAT	0.560	<b>0.730</b>	<b>0.612</b>	<b>3.666</b>	0.933
	5	TT	0.555	0.728	<b>0.615</b>	3.660	0.930
		NAT	<b>0.535</b>	<b>0.736</b>	0.612	<b>3.681</b>	<b>0.937</b>
60	15	TT	0.514	0.743	0.631	3.804	0.931
		NAT	<b>0.488</b>	<b>0.755</b>	<b>0.636</b>	<b>3.814</b>	<b>0.939</b>
	31	TT	<b>0.502</b>	0.748	<b>0.639</b>	<b>3.887</b>	0.931
		NAT	0.503	<b>0.750</b>	0.632	3.774	<b>0.938</b>

Table 5: Saliency quality metrics on DIEM testing set, for TASED Net, training with  $\text{KLD}-0.1\text{CC}-0.1\text{NSS}$  as discrepancy, and various number of training videos and observers. The best metrics between TT (Eq. 2 in main paper) and NAT are in bold.

### 5.3 DNN architectures

To further verify that NAT works effectively on different DNN architectures, independently from the adopted dataset, we train SalEMA [9] on the ForGED dataset. We use KLD as the discrepancy function, with RMSprop as the optimizer with a learning rate equal to  $1e^{-5}$  rather than Adam with learning rate  $1e^{-7}$  and binary cross entropy as discrepancy function, as suggested by the authors (an analysis of this hyperparameter choice is discussed later). Consistently with the other cases analyzed here, NAT outperforms TT, notably when the number of observers or videos is limited (Table 6).

train videos $V$	train obs. $N$	loss	KLD↓	CC↑	SIM↑	NSS↑	AUC-J↑
30	5	TT	1.229	0.546	0.412	2.911	0.912
		NAT	<b>1.187</b>	<b>0.559</b>	<b>0.428</b>	<b>3.050</b>	<b>0.915</b>
	15	TT	1.214	0.544	0.420	2.972	0.916
		NAT	<b>1.184</b>	<b>0.563</b>	<b>0.426</b>	<b>3.152</b>	<b>0.916</b>
100	5	TT	1.077	<b>0.600</b>	0.444	3.273	0.923
		NAT	<b>1.071</b>	0.599	<b>0.447</b>	<b>3.274</b>	<b>0.926</b>
379	2	TT	<b>1.054</b>	<b>0.601</b>	<b>0.447</b>	3.248	0.926
		NAT	1.076	0.600	0.440	<b>3.284</b>	<b>0.930</b>
	5	TT	<b>1.014</b>	0.623	<b>0.482</b>	<b>3.533</b>	0.929
		NAT	1.019	<b>0.623</b>	0.471	3.526	<b>0.930</b>

Table 6: Saliency quality metrics on ForGED testing set, for SaleMA, training with KLD as discrepancy, and various number of training videos and observers. The best metrics between TT (Eq. 2 in main paper) and NAT are in bold.

## 6 Additional training details (mentioned in Sec. 5)

Here we discuss more training additional training details for TASED-Net [10], SaleMA [9], and EML-Net [8]. The details of training ViNet are already mentioned in the main paper, Sec. 5. For all models, the code released by authors was used, with changes to reflect the new hyperparameter settings, specifying NAT loss function and faster data loading.<sup>1</sup> For all experiments, the training was stopped when the validation discrepancy does not improve for 10,000 iterations. All testing was performed at the original resolution for videos of all datasets: when the predicted output size is different, the predicted saliency maps were resized to original resolution.

**Hyperparameters for TASED training on LEDOV.** To ensure a fair comparison against traditional training and guarantee that the best performance is achieved for the given architecture and dataset, we first perform some hyperparameter tuning of TASED on LEDOV with traditional training (Eq. 2 in main paper). We found that using RMSprop with a learning rate of 0.001 for KLD optimization gives better performance than the default settings originally proposed for training on DHF1K (*i.e.*, SGD with momentum 0.9 and learning rate 0.1 for decoder stepping down by a factor of 0.1 at iteration 750 and 950, and 0.001 for encoder), as shown in Table 7 and in Fig. 5. Thus, we adopt RMSprop with a learning rate of 0.001 to train TASED for both traditional training and NAT in all the experiments. An exception to this rule is the traditional training with SGD reported in Table 3, where we adopt SGD with a learning rate of 0.0001 (any higher leads to training instabilities due to data noise) and momentum 0.9.

hyperparameter settings	KLD↓	CC↑	SIM↑	NSS↑	AUC-J↑
TASED-Net, SGD, learning rate schedule (default)	1.104	0.554	0.452	2.536	0.828
TASED-Net, RMSprop, 0.001, KLD (improved)	<b>0.754</b>	<b>0.724</b>	<b>0.572</b>	<b>4.227</b>	<b>0.921</b>

Table 7: Performance on LEDOV for TASED trained traditionally using KLD with original settings, and those used in the main paper (RMSprop, learning rate 0.001) on the full LEDOV training set. We adopted the best hyperparameter setting (best metrics in bold) for all experiments. \*Original settings: SGD, initial learning rate 0.1 for decoder and 0.001 for encoder, momentum 0.9.

<sup>1</sup>ViNet: <https://github.com/samyak0210/ViNet>, TASED-Net: <https://github.com/MichiganCOG/TASED-Net>, SaleMA: <https://github.com/Linardos/SaleMA>, EML-Net: <https://github.com/SenJia/EML-NET-Saliency>

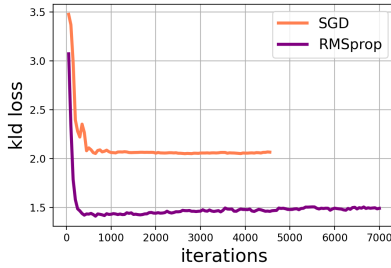


Figure 5: Validation-set performance plots (KLD vs. training iterations) for the LEDOV dataset during training of TASED with KLD as loss function and LEDOV dataset using: SGD, with default setting provided by authors; and RMSprop, learning rate 0.001. Based on this experiment, we choose RMSprop with a learning rate of 0.001 for our experiments.

**Hyperparameters for SalEMA training on LEDOV.** We train SalEMA [9] on the full LEDOV dataset with the default choice for loss function and optimizer (Adam optimizer, binary cross entropy, with learning rate  $1e^{-7}$ ), and compare against the adoption of the RMSprop optimizer with KLD as the loss function and 2 learning rates:  $1e^{-5}$  and  $1e^{-7}$  (see Table. 8). We train with LEDOV training set and we choose the best hyperparameter setting based on the LEDOV test-set performance for all of the experiments in the paper.

hyperparameter settings	KLD↓	CC↑	SIM↑	NSS↑	AUC-J↑
Adam, $1e^{-7}$ , BCE (original)	1.238	0.511	0.412	2.426	0.894
RMSprop, $1e^{-7}$ , KLD	1.206	0.532	0.418	2.602	0.900
RMSprop, $1e^{-5}$ , KLD	<b>1.052</b>	<b>0.612</b>	<b>0.463</b>	<b>3.237</b>	<b>0.912</b>

Table 8: Performance comparisons on LEDOV test set for SalEMA trained with the original hyperparameter settings and the ones used in this paper (RMPprop optimizer with  $1e^{-5}$  learning rate) after training on LEDOV training set. Best metrics are in bold.

**Details of training EML-Net.** We train EML-Net [9] for image-saliency on our noisy version of SALICON train set [9] (generated by randomly selecting a subset 5 or 15 fixations per image, see Sec. 6 in main paper). To do so, we select the ResNet50 backbone [9]. Consistent with recommendations from authors, we train two versions of the encoder: first, we finetune starting from ImageNet-pretrained weights [12], and second, we finetune from Places365-pretrained weights [16]. The two saliency models obtained from the encoder-training stage are input to the decoder-training pipeline to give the final image-saliency predictor for EML-Net approach. We adopt the EML discrepancy (which is a combination of KLD, CC and NSS losses described by authors) for training both traditionally (Eq. 2 in main paper) and using NAT. After searching through learning rates and optimizers, we find the author-specified choices to be most optimal: SGD with momentum with a learning rate of 0.01 at the beginning and multiplied by 0.1 after every epoch. We train both encoder and decoder for 10 epochs. After training, the best model for each experiment in Table 6 of the main paper is selected based on validation-set performance (using *all* available fixations on images in validation set), and submitted to SALICON benchmark for evaluation on the test set [9]. Note that even though the training is performed on few-fixation images to simulate a noisy version of the SALICON dataset, the evaluation on test set and validation set contains *all* of the available fixations.



## 7 Alternative methods to estimate $\tilde{x}$ (mentioned in Sec. 6)

In Sec. 6 of main paper, we discuss an alternative strategy using Gaussian kernel density estimation (KDE) with uniform regularizer to estimate  $\tilde{x}$  for training, instead of the common practice of blurring human gaze fixation locations using a Gaussian blur kernel of size approximately  $1^\circ$  viewing angle. We provide further details here. We estimate the optimal KDE bandwidth for *each* video frame, mixed with a uniform regularizer whose coefficient is also a parameter to be estimated. We do a per-frame estimation of optimal KDE bandwidth and mixing coefficient, to account for the general case where each frame can have a different variety of points of interest to attract gaze which cannot be explained with the optimal KDE bandwidth of another frame. The alternative to this is to estimate an optimal KDE bandwidth independent of the video frames, which amounts to the case of obtaining a universal Gaussian-blur kernel of a different size. In this case, the treatment of the underlying gaze data for obtaining the measured saliency maps,  $\tilde{x}_i$ , remains the same, in principle, as our experiments with  $\sim 1^\circ$  viewing-angle Gaussian-blur kernel (which amounts to 36 pixels and  $1920 \times 1080$  resolution for ForGED). To demonstrate this for completeness, in Table 9, we show some of the results for TASED trained with ForGED and KLD as discrepancy. For this experiment, the training gaze maps are estimated using a Gaussian-blur kernel of size 27 pixels (at resolution  $1920 \times 1080$ ), which amounts to  $\sim 0.75^\circ$  viewing angle. We note in Table 9 that NAT outperforms traditional training, consistent with our experiments with  $\sim 1^\circ$  viewing-angle Gaussian-blur kernel reported in the main paper.

train videos $V$	train obs. $N$	loss	KLD $\downarrow$	CC $\uparrow$	SIM $\uparrow$	NSS $\uparrow$	AUC-J $\uparrow$
30	2	TT	1.586	0.471	0.329	2.832	0.878
		NAT	<b>1.387</b>	<b>0.546</b>	<b>0.378</b>	<b>3.153</b>	<b>0.879</b>
	5	TT	1.358	0.563	0.345	3.184	0.903
		NAT	<b>1.239</b>	<b>0.565</b>	<b>0.406</b>	<b>3.272</b>	<b>0.905</b>
	15	TT	1.056	<b>0.622</b>	<b>0.483</b>	3.682	0.902
		NAT	<b>1.035</b>	0.616	0.476	<b>3.757</b>	<b>0.917</b>
100	5	TT	1.085	0.634	0.464	<b>3.770</b>	0.903
		NAT	<b>1.018</b>	<b>0.636</b>	<b>0.474</b>	3.633	<b>0.926</b>
379	5	TT	0.959	0.651	0.480	3.652	<b>0.931</b>
		NAT	<b>0.888</b>	<b>0.670</b>	<b>0.517</b>	<b>4.091</b>	0.924

Table 9: Performance comparisons on ForGED test set for TASED trained with KLD as discrepancy. Instead of computing gaze maps for train set with Gaussian blur kernel of size approximately  $1^\circ$  viewing angle (which amounts of 36 pixels at  $1920 \times 1080$  resolution), we use a Gaussian blur kernel of size approximately  $0.75^\circ$  viewing angle (27 pixels). As we can see, the conclusion regarding the superior performance of NAT compared to traditional training applies independent of blur kernel size.

To estimate the optimal bandwidth using KDE, we optimize a gold-standard model for saliency prediction, which predicts the probability of fixation for one observer, given the gaze data from the remaining observers for the video frame (leave-one-out cross-validation) [8, 13]. We observe that, when gaze fixation locations are sparsely distributed across a frame, the optimal bandwidth for KDE is high, which would result is high-spread, almost-uniform saliency maps. Independent of the estimation strategy for  $\tilde{x}$ , we posit that there is an underlying uncertainty / noise in the measured saliency map – which is accounted for during training using NAT, to obtain improved performance over traditional training.



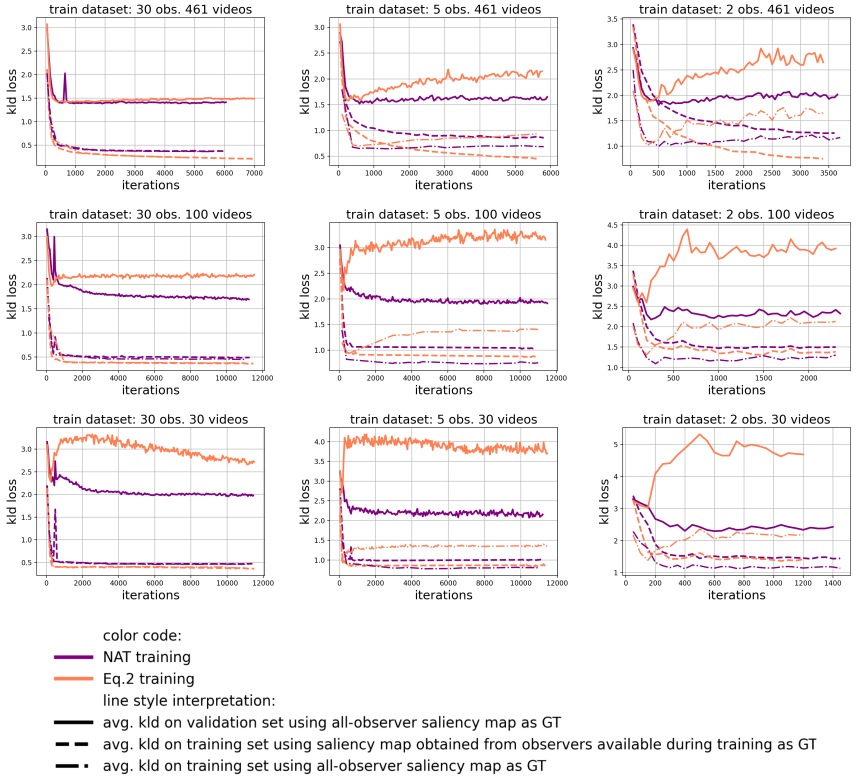


Figure 6: Training-set and validation-set KLD as a function of training iterations for TASED trained on LEDOV (“GT” in the legend indicates “ground-truth”). In contrast to the traditional training (Eq.2 in main paper), NAT does not overfit.

## 8 Overfitting behavior with NAT (mentioned in Sec. 3)

Figure 6 shows the training and validation set performance (in terms of KLD) as a function of the training iteration when training TASED on LEDOV dataset with KLD discrepancy, for different number of observers and videos in the training set. For both the traditional approach (dashed orange line) and NAT (dashed purple line), the training-set curves decrease regularly, as expected in a smooth optimization process. However, the validation-set curves for traditional training (continuous orange line) quickly reach a minimum and then start diverging towards a higher asymptotic value, which is a clear sign of overfitting. On the other hand, the validation curves for NAT (continuous purple line) are always lower (suggesting better performance) and tend to stabilize around asymptotic values without growing anymore — a clear sign, in this case, that overfitting is avoided. Note that for the training-set curves (dashed lines), the human saliency map used for KLD computation is derived using the limited number of observers available during the specific training experiment. As an additional check for the overfitting behavior of traditional training, we plot the performance of training set when compared against human saliency maps obtained from *all* the observers available in the training videos (32). These are indicated with dash-dotted lines. For few-observer experiments, the performance of traditional training on all-observer evaluations gets worse with increasing iterations. On the contrary, the performance on validation set, training set,

and all-observer training set do not generally show signs of overfitting for NAT. Only in few cases, NAT plots are unstable at the beginning of the training (see the peaks in the validation curves in the left most panels for 30 observers trainings), but then the curves stabilize to an asymptotic value. The only exception to this is represented by the upper right panel in the figure (2-observer training with 461 videos), where we believe that the slight increase in the validation-set performance value is due to the approximation introduced in NAT to make it computable in practice. We observed a similar behavior when training on other datasets.

## References

- [1] Richard Droste, Jianbo Jiao, and J. Alison Noble. Unified Image and Video Saliency Modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 2016.
- [3] Sen Jia and Neil DB Bruce. Eml-net: An expandable multi-layer network for saliency prediction. *Image and Vision Computing*, 2020.
- [4] Lai Jiang, Mai Xu, Tie Liu, Minglang Qiao, and Zulin Wang. DeepVS: A deep learning based video saliency prediction approach. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.
- [5] Lai Jiang, Mai Xu, Zulin Wang, and Leonid Sigal. DeepVS2.0: A saliency-structured deep learning method for predicting dynamic visual attention. *International Journal of Computer Vision (IJCV)*, 2021.
- [6] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [7] Tilke Judd, Frédo Durand, and Antonio Torralba. A benchmark of computational models of saliency to predict human fixations. 2012.
- [8] Matthias Kümmerer, Thomas SA Wallis, and Matthias Bethge. Information-theoretic model comparison unifies saliency metrics. *Proceedings of the National Academy of Sciences*, 2015.
- [9] Panagiotis Linardos, Eva Mohedano, Juan Jose Nieto, Noel E O’Connor, Xavier Giro-i Nieto, and Kevin McGuinness. Simple vs complex temporal recurrences for video saliency prediction. *arXiv*, 2019.
- [10] K. Min and J. Corso. TASED-Net: Temporally-aggregating spatial encoder-decoder network for video saliency detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.
- [11] Parag Mital, Tim Smith, Robin Hill, and John Henderson. Clustering of gaze during dynamic scene viewing is predicted by motion. *Cognitive Computation*, 2011.

- [12] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 2015.
- [13] Matthias Tangemann, Matthias Kümmerer, Thomas S.A. Wallis, and Matthias Bethge. Measuring the importance of temporal features in video saliency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [14] W. Wang, J. Shen, F. Guo, M. Cheng, and A. Borji. Revisiting video saliency: A large-scale benchmark and a new model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [15] Niklas Wilming, Torsten Betz, Tim C Kietzmann, and Peter König. Measures and limits of models of fixation selection. *PloS one*, 2011.
- [16] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017.